Woods Hole Oceanographic Institution
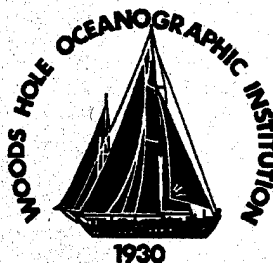

HydroBase

A Database of Hydrographic Stations and Tools for Climatological Analysis

by

Ruth Gorski Curry

March 1996

Technical Report

Approved for public release; distribution unlimited.

WHOI-96-01

HydroBase
A Database of Hydrographic Stations and Tools
for Climatological Analysis

by

Ruth Gorski Curry

Woods Hole Oceanographic Institution
Woods Hole, Massachusetts 02543

March 1996

Technical Report

Approved for Distribution:

Philip L. Richardson, Chair
Department of Physical Oceanography

Table of Contents

Abstract

This report documents the organization, functionality, and algorithms of a software package which operates as a database manager and toolset for climatological analysis of hydrographic station data. It details the methods of quality control used in construction of the small, but growing, database and discusses some of the improvements *HydroBase* methods offer over existing gridded databases, including a short comparison to Levitus's *World Ocean Atlas 1994* package. A large portion of this technical reference is devoted to describing the software modules and providing examples for their use.

I. What Is *HydroBase*?

HydroBase is a set of utilities designed to enable the user to access and manipulate profiled station data plus a limited, but growing, database of hydrographic and CTD stations. It is *not* a graphics package. The products of *HydroBase* are ASCII or netCDF files which the user can input to his/her own favorite graphics software. The source code for these utilities is also provided so that individuals can build their own applications from the basic modules. The strengths of this package lie in its flexibility to produce gridded datasets on scales specified by the user and its ability to handle any number of measured or derived properties. Its most important feature, perhaps, and one which distinguishes this from other databases is its implementation of isopycnal averaging. Although most other gridded datasets are produced by averaging properties on standard depth surfaces, *HydroBase* performs its gridding on closely spaced density surfaces and then transforms the averaged matrix from density to depth coordinates to ultimately produce a three dimensional matrix gridded in latitude, longitude and depth. Recent analysis by Lozier, McCartney, and Owens (1994) demonstrates that averaging on depth or pressure surfaces produces water properties that are dissimilar to the observed water masses; and that these artifacts are avoided by averaging/gridding in density coordinates.

The *HydroBase* utilities permit the user to :

- extract station subsets from the database using a variety of parameters;
- compute hydrographic properties from observations;
- produce 3-D gridded datasets of any hydrographic properties;
- specify all gridding parameters (latitude, longitude, depth, time);
- project hydrographic properties onto surfaces;
- smooth the gridded data on a surface;
- analyze data by groups of years.

The source code includes libraries to facilitate the reading and writing of *HydroBase* files and the computation of hydrographic properties for C programmers who wish to utilize the data in other ways.

The style of these programs was patterned after the GMT-SYSTEM[1] (Wessel and Smith, 1991), a popular and freely distributed graphics mapping package, because it has proved to be relatively easy to understand and use. In addition to its graphics capabilities, GMT provides several modules for interpolating and low-pass filtering 2-dimensional gridded data which are particularly useful in the context of filling data holes in hydrographic property fields. I have achieved excellent results with their module, *surface*, which implements a minimum

---

1.At this writing, GMT-SYSTEM is available by sending email to listserver@soest.hawaii.edu containing the single message: information gmtgroup.

curvature spline under tension algorithm (Smith and Wessel, 1990). Rather than reinventing the wheel and writing a *HydroBase* module to perform the same task, I recommend making use of this tool (or other statistical tools) to accomplish "objective mapping" of the property fields when necessary. Several script files have been included in the ./extras subdirectory of the software distribution which use GMT to produce plots in *PostScript* format from the *HydroBase* files.

A. Data Sources and Organization

The station data used in this synthesis were obtained from the National Oceanographic Data Center (NODC) and represent all hydrographic station data (pressure, temperature, salinity, and oxygen) available as of April 1990. At the time this document is being produced, only one ocean basin has been included into *HydroBase:*

North Atlantic (0° - 75° N, 85° W - 20° E) : 131635 stations

As time and funding permit, the South Atlantic, Pacific and Indian Ocean basins will be added .

The stations are arranged geographically by 10-degree WMO (Marsden) squares and stored in files named *msq10.extent*, where *msq10* is the 4-digit WMO Square designation:

$$WMO \; Square = hem * 1000 + (lat/10) * 100 + (lon/10);$$

where hem = 1 or 7 for Northern hemisphere, (see diagram below)

3 or 5 for Southern hemisphere,

1 or 3 for Eastern hemisphere,

5 or 7 for Western hemisphere;

lat = (integer) latitude,   lon = (integer) longitude
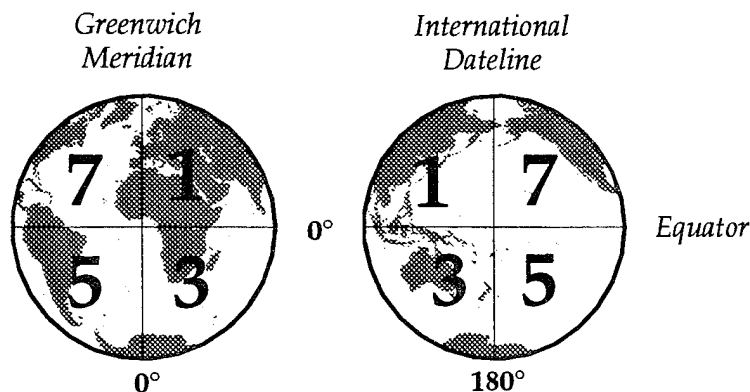
and *extent* denotes the type of data in the file, which can be any combination of:

.qc   : quality controlled

.shall : stations < 200 meters depth

.ctd   : CTD high resolution data

.raw : original data (not quality checked)

There is no particular ordering of stations within each file. See Appendix A for information on the structure of a *HydroBase* station.

The original database (Lozier, Owens, and Curry., 1996) contained only hydrographic stations >200 meters water depth. Subsequently, the coastal data, other shallow stations, and some CTD sections have been put into *HydroBase* format. These files are designated *msq10*.shall and *msq10*.ctd. Because *HydroBase* will continue to grow and evolve with time, a README file is stored with the data to log all updates and changes to the database.

B. Quality Control

Since producing a dataset suitable for climatological study was central to the goals of this project, the methods of quality control were designed to identify and eliminate unrealistic data without destroying the natural variation inherent to ocean circulation. The stations which successfully passed through our quality control steps may not meet everyone's standards because quality control is a very subjective process. Nor do we claim to have identified every "bad" station or observation and kept every "good" one. These quality control methods are best viewed as a preliminary process to deal with substantial numbers of unrealistic values scattered throughout a dataset of enormous size. Your usage of this dataset will dictate the need for additional examination of individual data points.

1. Range Checking and Missing Data

A preliminary sieve of the data eliminated values outside of broad property ranges which were defined for temperature, salinity and oxygen as a function of depth and latitude using atlases and synoptic section for guidance. Because subsequent quality control steps would require the ability to compute potential density, any observation level missing either temperature or salinity was entirely eliminated. Temperature and salinity were retained for any level missing only an oxygen value.

2. Statistical Checking

To further identify questionable data points, we used the fact that potential temperature (θ) - salinity (S) and θ - oxygen (O_2) relationships are locally well defined in the world oceans. Plots of θ-S and θ-O_2 for all stations within a discrete geographic area illustrate these relationships and graphically reveal data points that deviate significantly from them. These plots are often used to identify erroneous points visually in data as they are collected and analyzed. Because hand-picking outliers from visual representations of several hundred thousand stations was not a viable option, a method was devised to identify these outliers computationally.

The θ-S curve for a group of stations with similar property profiles can be approximated by vertically subdividing the observed points into smaller groups as a function of density and then connecting the mean θ,S point for each density bin with straight lines (figure 1). As the number of density bins increases, these line segments converge on the shape of the θ-S curve. A line tangent to this curve at the mean θ,S point of each density bin approximates the θ-S relation for that bin. The standard deviation of salinity as a function of potential temperature can be easily computed as the least squares distance from the tangent line to the corresponding observed θ,S combination. Figure 1 illustrates that a distance of approximately ± 2 standard deviations from the tangent line adequately characterizes the natural spread of observations; most points outside of this envelope appear to be errant. Although density variations are often dominated by potential temperature, in some locations, notably the subpolar oceans and near the sea-surface, salinity has a greater influence. In these instances, it is more sensible to compute the standard deviation of potential temperature as a function of
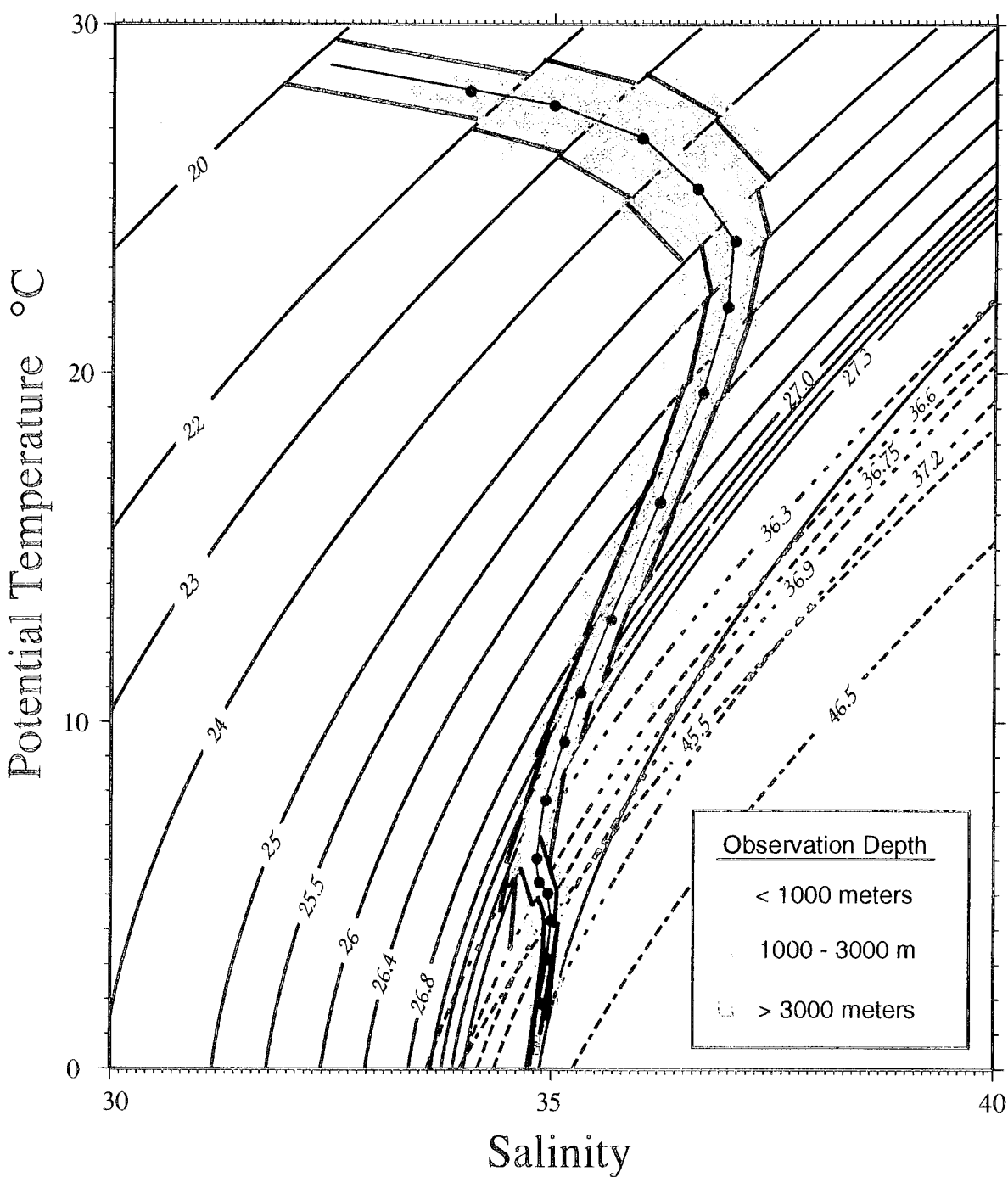
Figure 1: A θ-S diagram of actual observations in a 5° square geographical area illustrates the paradigm by which the data were statistically checked. The contours delineate the density bins used in the statistical fit (solid contours are σ relative to 0 db., dashed lines are σ-2000 and σ-4000 bins used for deeper observations). Different symbols denote three depth regimes which determined the reference level associated with each point. Filled circles depict the mean θ-S pair for each bin. A thin solid line connecting the circles approximates the mean θ-S curve for the area. The heavy, broken lines define the 2.3 standard deviation envelope outside of which points were eliminated from the database.

salinity. Similar characterizations can also be made for the θ-O_2 (or S-O_2) relation, where the standard deviation of oxygen observations is computed as function of potential temperature (or salinity when it varies more rapidly than temperature).

In order to use these statistical characterizations to identify outliers, the data were first subdivided into geographic areas to achieve a balance between two goals: (1) to create groups with similar θ-S and θ-O_2 profiles; and (2) to maximize the number of stations in each group. Using all station data within a 5-degree square area, θ-S and θ-O_2 plots were produced. In areas which showed a large degree of scatter in the property relationships, 2.5-degree subdivisions were made. If this subdivision was inadequate to obtain relatively tight property-property relationships, the data were further subdivided into 1-degree squares. Where the station density was low, neighboring 1-degree squares were combined into larger areas to achieve a minimum of ten stations per area with which to establish statistically significant mean property relations.

We used the smaller geographic areas (1 - 2.5 degree) to compute mean property relations for the upper 1000 meters only. Below 1000 meters, larger geographic areas (5-degree squares) compensated for the decrease of available observations and enabled us to better identify deviant points by more precisely defining the mean property relations. We found that this generally precluded an aberrant cast from skewing the mean and standard deviation enough to slip through the statistical checking.

The data were also divided into vertically contiguous sections for which the mean θ-S and θ-O_2 relations were approximately linear. We opted to subdivide the data into density bins rather than by depth and chose these bins empirically by superposing isopycnals onto the θ-S diagrams and examining the resultant segments of the θ-S profiles. After experimenting with these diagrams in various parts of the world oceans, we arrived at sets of density bins appropriate to generalized geographic areas.

After establishing geographic and vertical subdivisions, the remainder of the procedure was straightforward. For each geographic area, we first computed the slope, intercept, and appropriate standard deviation for the lines characterizing the θ-S and θ-O_2 relationships. (In bins where the ratio Δsalt/Δtheta exceeded an arbitrarily chosen value of 1.0, salinity was used as the independent parameter in evaluating the standard deviation.) Then each temperature, salinity, and oxygen observation was individually compared to the statistical characterization in the appropriate density bin. Any value which differed by more than 2.3 standard deviations[2] from the line was eliminated. If a θ-S pair fell outside this envelope, the entire scan was eliminated because we would later need both properties to compute density. Any oxygen point falling outside the envelope was eliminated, but the temperature and salinity observations were retained. Any point whose density fell into none of the defined bins or into a bin containing less than three points was eliminated. Finally, any station having greater than 20% bad observations was completely eliminated.

This procedure was applied twice to the database. In the North Atlantic, the first pass eliminated 7792 (5.5%) stations. Temperature-salinity (oxygen) outliers were identified in 6.6% (12.8%) of the individual scans. The higher oxygen elimination reflects the removal of even a good oxygen point if its associated temperature or salinity failed the test. For the second pass, the data which emerged from the first check were used to recompute the means and standard deviations for the property relations. An additional 1.3% of stations were eliminated. For this pass, 2.3% of individual samples failed to meet the θ-S criterion, while 4.5% of oxygen samples were eliminated. Overall, the statistical check reduced the North Atlantic dataset by 6.6%, to a total of 132,608 stations. Finally, we note that this data elimination was distributed uniformly over the water column. Figure 2 shows that the vertical distribution of observations before and after quality control are virtually identical indicating that the shape of the vertical distribution is a result of sampling strategy and not an artifact of our quality control procedures.

---

2. This limit was chosen on the statistical basis that approximately 98% of all observations in a normally distributed population will fall within 2.3 standard deviations on either side of the mean.

Percent of Stations with Observations

(Actual Number of Stations where echo sounder depth > standard depth)

	0	50	100		0	50	100

Standard Depth (meters)

Depth	Number
0	131635
10	125887
20	124687
30	124204
50	121771
75	121508
100	121277
125	121049
200	120947
250	113546
300	108540
350	100694
400	95562
500	89960
600	86658
700	84128
800	81724
900	79457
1000	76627
1100	74719
1200	72964
1300	71449
1400	69928
1500	65941
1750	62510
2000	54607
2500	46841
3000	35049
3500	23508
4000	16442
4500	6888
5000	1294
5500	253
6000	131
6500	101
7000	78
7500	45
8000	20
8500	

P - T - S

Ox

■ Before QC
▧ After QC

Figure 2: Vertical distribution of observations in North Atlantic dataset from Lozier *et al.* (1996) (only stations > 200 meters are included here) before and after quality control procedures. Each bar depicts the percentage of stations in which at least one observation was made between the standard depths listed on the y-axis. This percentage is based only on the number of stations where the sea floor is deeper than the standard depth.

3. Oxygen Quality

The method for titrating oxygen samples changed in the late 1950's from a potassium dichromate standard to a bi-iodate standard. Worthington (1976) suggests that all data produced under the old standard should be multiplied by 1.048 to correct for inconsistencies generated by the change in methods. We reluctantly opted to eliminate older oxygen data (pre-1960) completely from the database rather than attempt to adjust those values.

4. Further Quality Control

Contoured fields of the standard deviations computed for the means of each property proved an invaluable tool in identifying some remaining questionable values. In several instances data from a single cruise with many stations adversely biased the mean in a particular area creating a large standard deviation which enlarged the envelope of allowable observations in the statistical check. In these cases, we manually identified and removed the suspect cruise from the original set of stations, then completely recomputed the statistical check for that area.

C. Comparison to *World Ocean Atlas 1994* Database

Because the *Levitus* climatology (Levitus, 1982; Boyer and Levitus,1994; Levitus and Boyer, 1994; Levitus, Burgett, and Boyer, 1994) has become a standard in the oceanographic community, a comparison between the latest version, *World Ocean Atlas 1994* (hereafter *WOA*), and its 1-degree gridded dataset and a *HydroBase* 1-degree dataset is included here. Both datasets use the NODC archive of hydrographic stations to produce gridded values -- so the differences do not reflect an incongruity in the number of data available. This comparison is not intended to disparage the *Levitus* data -- which represents an enormous effort and has proven itself quite useful -- but rather to demonstrate that thoughtful adjustments to methods have brought about some measurable improvement to the climatology.

First, *HydroBase* methods more correctly approximate the observed θ-S relations -- especially in the upper 1000 meters. Figure 3 shows the potential temperature fields on a density surface near 600 meters depth in the subtropical gyre as computed from each dataset. The subtropical region of the *WOA* map describes numerous sets of closed contours -- warm patches oriented zonally along 27°N as well as near the axis of the Gulf Stream system, plus a cold anomaly in the center of the gyre -- which are absent from the *HydroBase* map.

To further probe these anomalies, figures 4 and 5 compare the depth, potential temperature and salinity relations for 1) the actual observed data, 2) the *WOA* 1° averaged values, and 3) the *HydroBase* 1° averaged values for two locations. In the warm, salty case (figure 4), the averaged *WOA* θ-S values are clearly distorted from the actual observed θ-S curve: higher in salinity for temperatures between 7-14°C while many fresher values prevail in the 17-22° C range. This geographic region, strongly influenced by the meandering position of the Gulf Stream, exhibits great variability in its temperature and salinity characteristics at depths <1000 meters. Lozier *et al.* (1994) demonstrate quite clearly that averaging on pressure surfaces (*WOA* method) in such a frontal area results precisely in this distortion of the θ-S relation, while isopycnal averaging (*HydroBase* method) more closely reproduces the observed θ-S curve.

The cold, fresh patch near the gyre interior also represents a shift in the θ-S relation of the *WOA* dataset relative to the observations (figure 5). In this case, salinities are slightly, but consistently, fresher between 12-17°C compared to the observed properties. The cause of this shift is not, however, isobaric averaging -- the water exhibits far less θ-S variability and isobarically averaging even the most diverse water parcels found at

any given depth does not produce the θ-S defined by the *WOA* curve. The explanation for this remains unclear; possibly the result of the smoothing or objective mapping procedure or it may reflect the inclusion of XBT data (Levitus, personal communication) which may have biased the temperature averages. While *HydroBase* exclusively employs θ-S pairs in forming averages, the *WOA* averages incorporate temperatures which are not balanced by salinities. The fact that the θ-S shift corresponds to depths < 700 meters (XBT depth range) lends support to this possibility.

HydroBase methods also enhance the resolution of surfaces, especially deeper than ~2000 meters. Because original observations are used, the averaging process can be tuned to resolve the property gradients at scales much finer than the 500-meter standard depth spacing onto which the *WOA* data is interpolated before averaging. *HydroBase* also preserves and utilizes the bottom observation of each cast, whereas this information is lost in the *WOA* method. Figure 6 depicts the positive effects of this approach by comparing pressure and potential temperature fields from each dataset on a deep density surface (σ-4000 = 45.907). Most notably missing from the *WOA* dataset are the deep western boundary currents which contribute a significant amount of shear to the *HydroBase* surface. Almost no *WOA* data show up north of 50°N; this is directly related to the loss of bottom information when the data is interpolated onto standard depths spaced 500 meters apart. Figure 7 illustrates how virtually none of the data below 5000 meters is incorporated into the *WOA* averaged dataset because the casts do not quite reach to the next standard depth, 5500 m; and that this further results in the omission of a significant section of the θ-S curve. In the north, consistent omissions of near-bottom data, like this example, translate to the disappearance of the deep boundary currents.

Important differences also exist in the potential temperature field of figure 6. West of 60°W, the *WOA* temperatures are too warm (and therefore salty) on this density surface. Figure 7 shows that the *WOA* averaged salinities at 3500, 4000 and 4500 meters are strongly skewed toward salty values in the western Atlantic near 28°N. Here again, it is not possible to determine the source of this disparity: quality control, interpolation onto standard depths, smoothing, or objective mapping are possible causes. Taken together, the pressure and temperature fields from the *WOA* dataset give a completely different view of the abyssal circulation than a contoured version of actual observations gives, e.g. Reid (1994). In contrast, *Hydrobase* replicates such abyssal fields -- with allowances for data coverage and smoothing -- because its averaging methods more closely reproduce the observed θ-S relations.
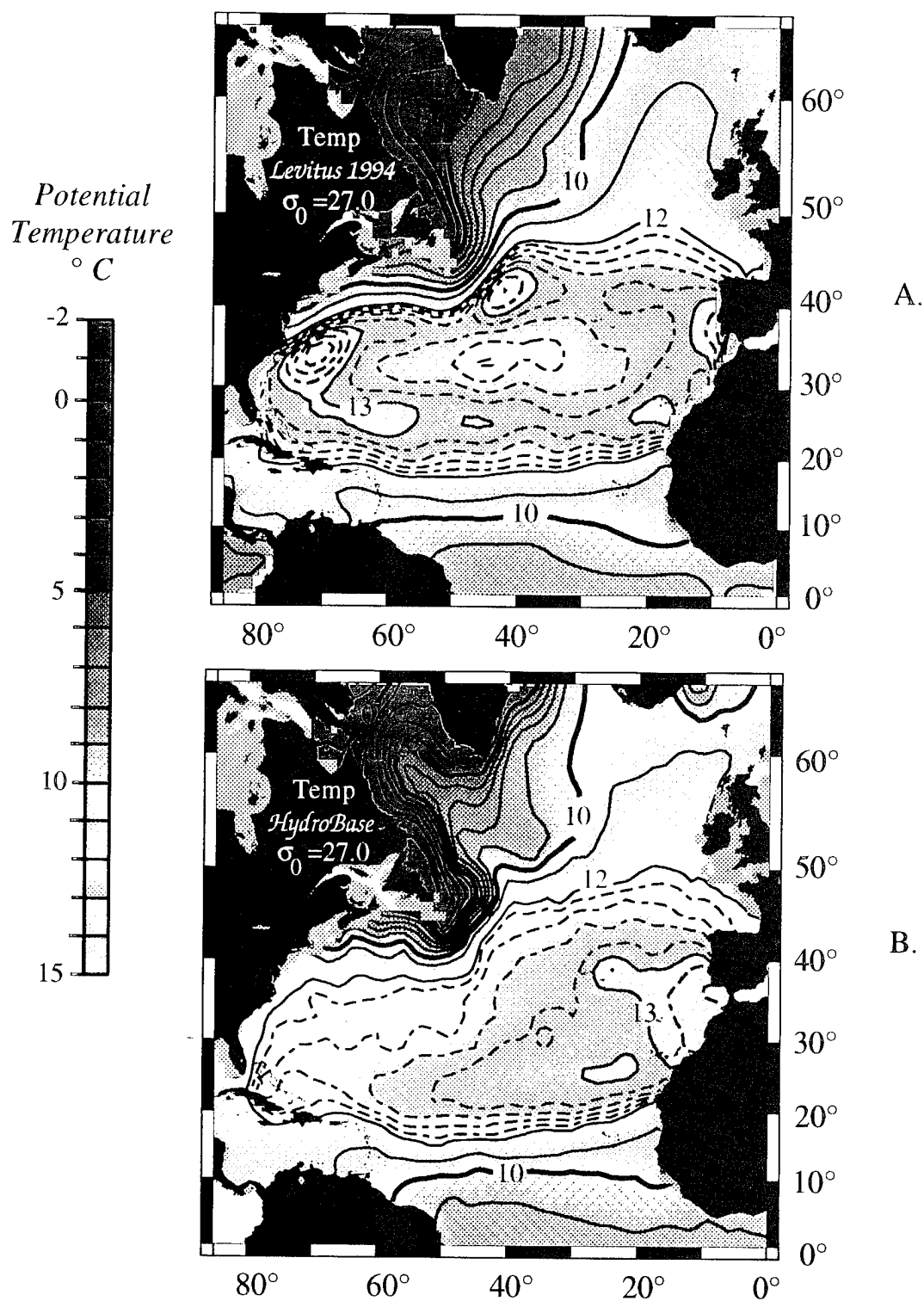
Figure 3: Potential temperature on a density suface in the thermocline of the subtropical gyre from A) *Levitus'* World Ocean Atlas 1994 database and B) *HydroBase*. Both represent the 1° gridded annual averages which were treated identically to produce these maps. The *WOA* map contains anomalously warm (red) patches along 28°N and in the Gulf Stream system, and anomalously cooler temperatures (closed yellow contours) in the center of the subtropical gyre. Isopycnal averaging is one factor in avoiding such anomalies.
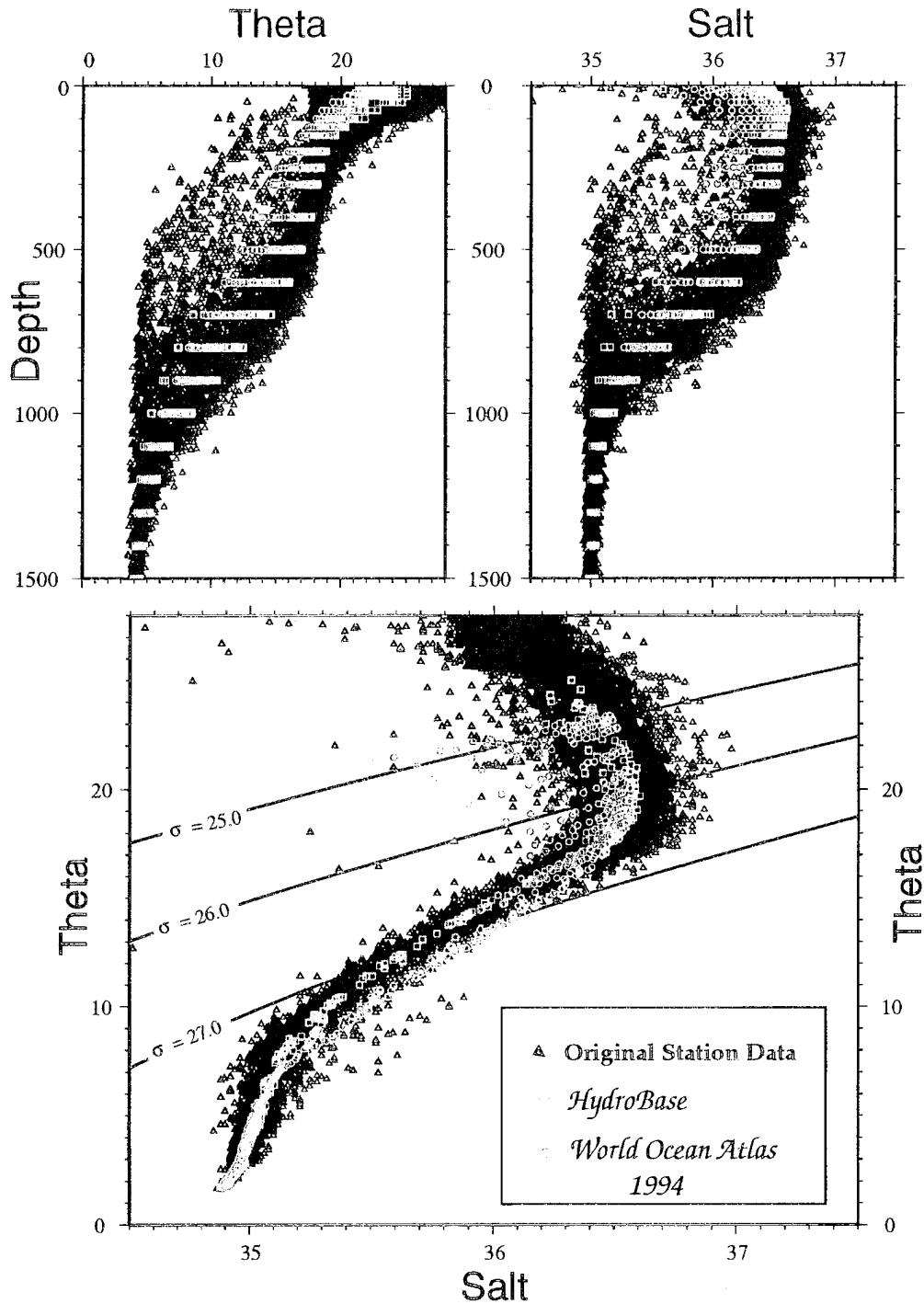
Figure 4: Temperature and salinity plotted vs. depth and in θ-S space illustrate how the 1° gridded values of *HydroBase* and *World Ocean Atlas 1994* (Levitus, 1994) compare to the original observations. The points represent all data from each dataset in a 4° square situated across the Gulf Stream near 35°N, 70°W. Three density contours have also been mapped into the θ-S diagram. The *WOA* θ-S (red) clearly deviates from the observed θ-S between 7-16°C and produces the warm, salty anomaly at density σ = 27.0 near the Gulf Stream shown on the map in figure 3. Isopycnal averaging (*HydroBase*, blue) more correctly approximates the observed θ-S relation.
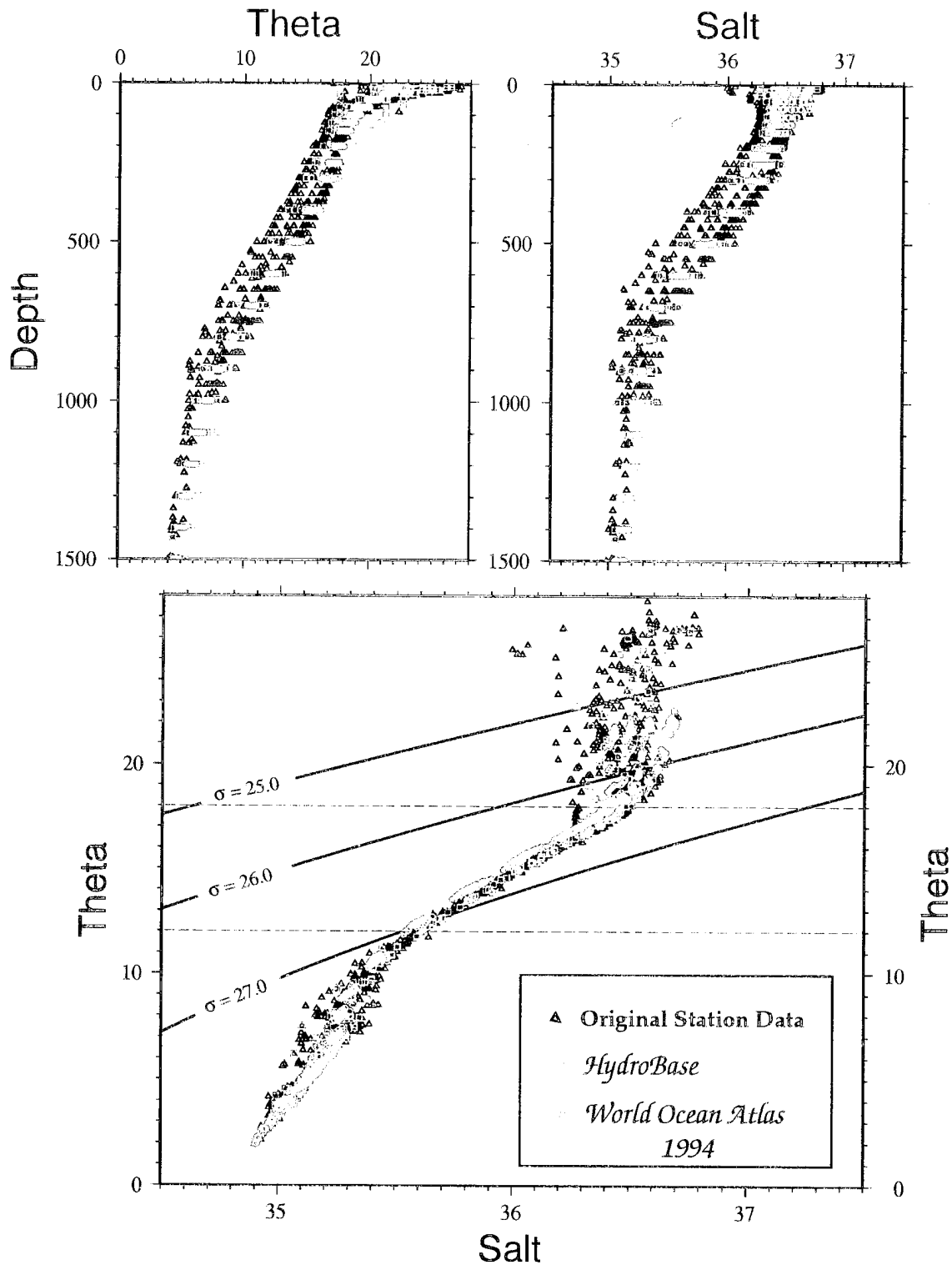
Figure 5: Potential temperature and salinity from a 4° square geographical area near 33°N, 42°W plotted vs. depth and in θ-S space compares 1° gridded values from *HydroBase* and *World Ocean Atlas 1994* (Levitus, 1994) to the original observations from which both were derived. Selected density contours are also mapped onto the θ-S diagram. The *WOA* θ-S is slightly fresher than both the observed data and the *HydroBase* averages at temperatures between 12-18°C. This results in a cold, fresh anomaly in the center of the subtropical gyre at density σ = 27.0 as shown in figure 3 (yellow patch).

A.

B.

C.

D.
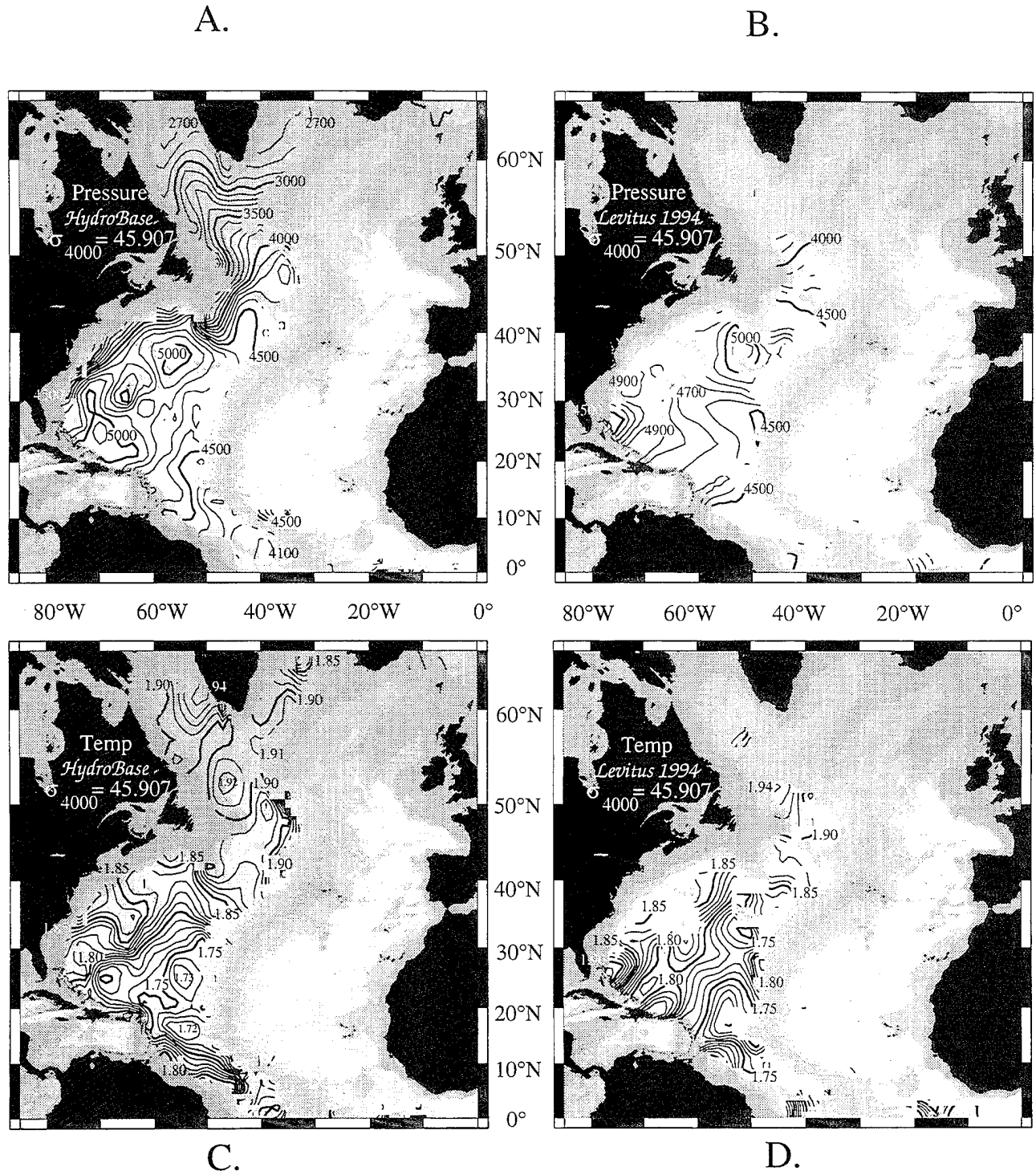
Figure 6: Properties on a deep density surface (σ_{4000} = 45.907) from 1° gridded datasets: A) *HydroBase* pressure; B) *World Ocean Atlas 1994* (Levitus, 1994) pressure; C) *HydroBase* potential temperature; and D) *World Ocean Atlas 1994* potential temperature. HydroBase methods permit better resolution of the deep western boundary currents and other features of the abyssal circulation.
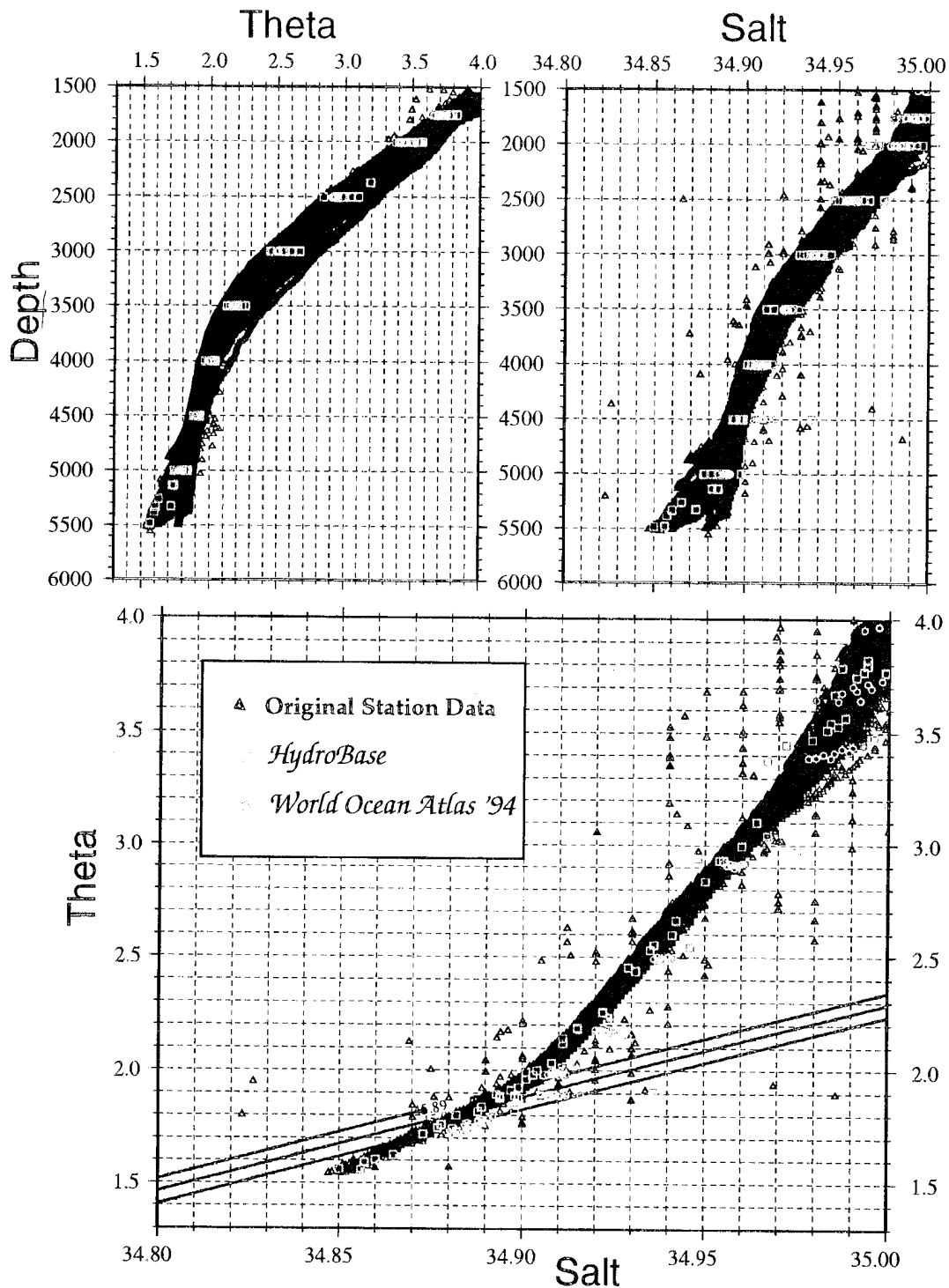
Figure 7: Potential temperature and salinity from a 5° square geographical area near 25°N, 72°W plotted vs. depth and in θ-S space compares averaged values from *HydroBase* and *World Ocean Atlas 1994* (Levitus, 1994) to the original observations from which both were derived. Selected density contours (45.89, 45.90, and 45.91) are also mapped onto the θ-S diagram. The *WOA* salinities are significantly higher than the observations near 4500 meters leading to a skewed θ-S curve. The WOA omits nearly all data > 5000 because most stations do not reach to the next standard depth level, 5500m. The result is that the deepest part of the θ-S curve is not represented at all in the WOA dataset.

II. HydroBase Utilities

A. An Overview

The *HydroBase* utilities consist of a set of executables, which you build on your platform, plus a set of C libraries which perform the basic functions necessary to read and write *HydroBase* files. The source code resides in the /src subdirectory of the software package in the following files:

<table>
<tr><td colspan="3" align="center"><u>Main Modules</u></td></tr>
<tr><td>cdfinfo</td><td>grid3d</td><td>propcalc</td></tr>
<tr><td>cdf2asc</td><td>gridsurf</td><td>smoothsurf</td></tr>
<tr><td>extract</td><td>invertblanks</td><td>surfdiff</td></tr>
<tr><td>findblanks</td><td>ms*sort</td><td>timeprop</td></tr>
<tr><td>getpos</td><td>preblank</td><td>xy4gmt</td></tr>
</table>

<table>
<tr><td><u>Libraries</u></td><td><u>Definitions/Structures</u></td></tr>
<tr><td>prop_subs.c</td><td>hydro_cdf.h</td></tr>
<tr><td>hydro_utils.c</td><td>hydrobase.h</td></tr>
<tr><td>hydro_cdf.c</td><td></td></tr>
<tr><td>phyprops.f</td><td></td></tr>
<tr><td>eos80d.f</td><td></td></tr>
</table>
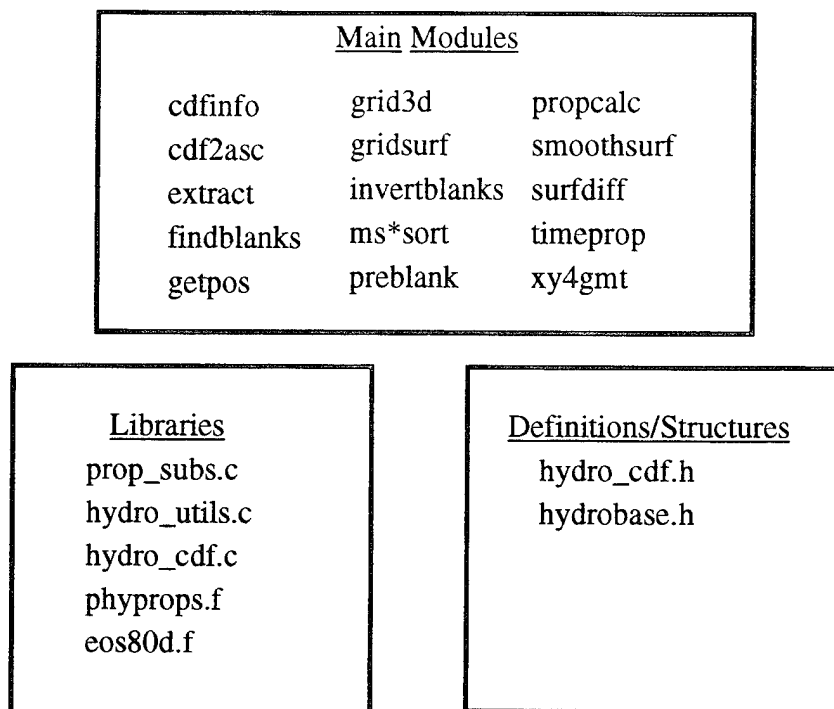
Each of the main modules is invoked from the command line with a set of arguments to specify various options. In general, the rules governing the syntax of the arguments are few:

> - the input files are listed as the first arguments -- i.e. immediately following the program name.
> - the structure of the arguments is generally:
>
>   **-O***value1/value2/value3*
>
>   where a dash precedes the uppercase Option letter, followed by the appropriate values, which are separated by slashes. No white space is permitted between any parts of a single argument.
> - In this document, optional arguments will be placed in square brackets, i.e. [**-O***value1/value2*]; required arguments will be listed without brackets.

Figure 8 lists the software modules available, what type of input/output files each reads/produces and how they are related to one another.

Figure 8: I/O diagram relating various HydroBase modules. The names of modules are highlighted with yellow while the arrows denote the types of input and output they operate on and produce. The blue pathway indicates a typical sequence of modules used to produce a property mapped on a horizontal surface.

Each module reads and produces a file format which falls into one of these categories:

HydroBase station files:

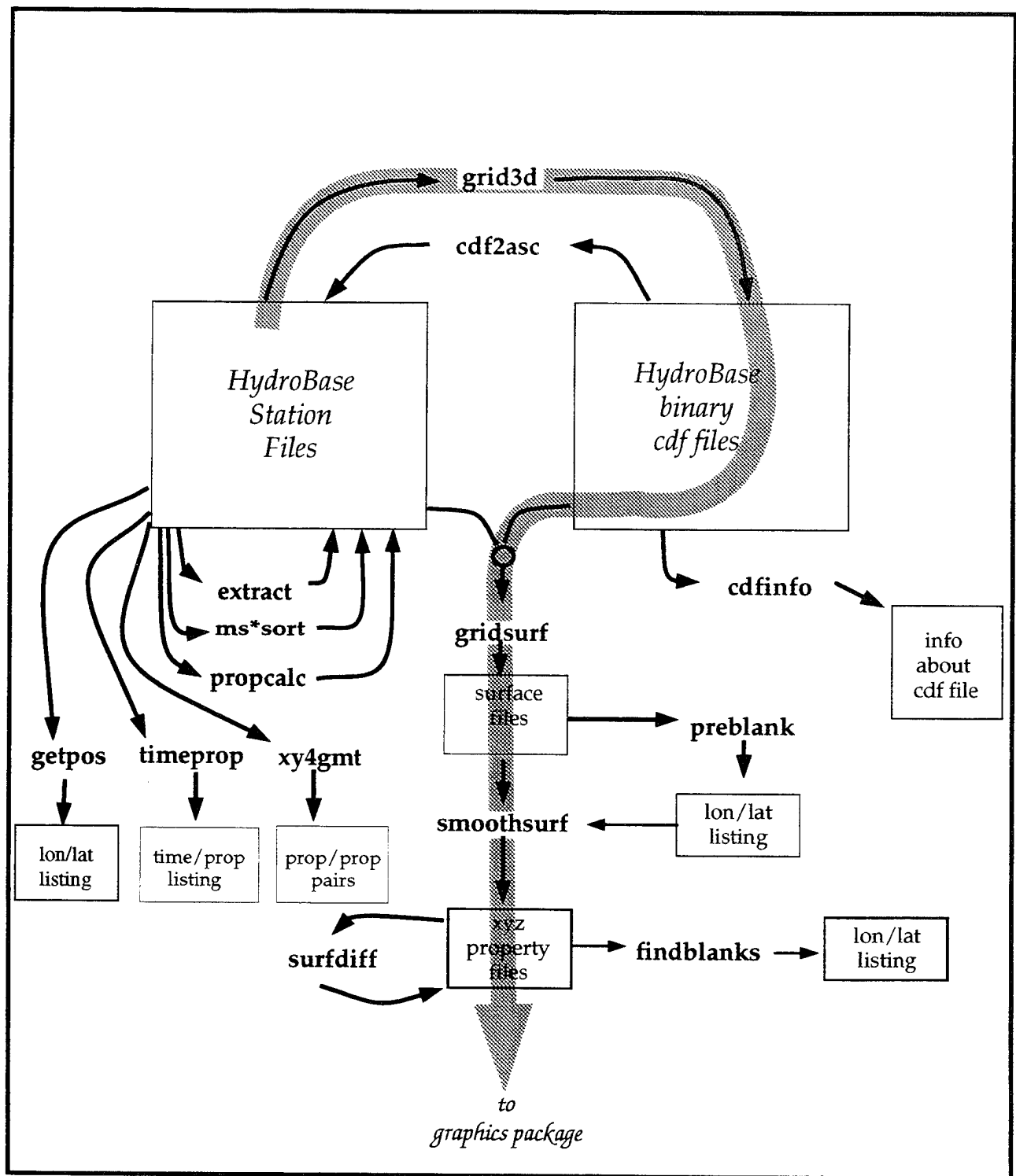> ASCII text consisting of a header line and variable number of data records in which original station data is stored.

binary netCDF "cdf_files":

> hydrographic properties gridded in 4-D as a function of latitude, longitude, depth, and time and stored in files conforming to the XDR protocol.

ASCII listings:

> output from surface gridding and smoothing programs, generally lists lat, lon and property values.

Some modules perform database management functions (*extract, ms*sort, propcalc, cdf2asc*), or produce listings (*getpos, timeprop, xy4gmt, preblank, findblanks, cdfinfo*). *Grid3d* produces a 3-dimensional matrix of isopycnally averaged properties. *Gridsurf* produces a 2-dimensional matrix of a property on some surface and can take either *HydroBase* station files or cdf files as input. When *HydroBase* station files are used, the averaging is done on the 2-dimensional surface specified. If that surface is a density surface, then the data will be isopycnally averaged, but if the surface is some other property (depth, pressure, temperature....) the averaging will be performed on that surface (i.e.isobarically, isothermally, etc... averaged). *Smoothsurf* takes the 2-dimensional surface and splits it into individual property files (lat-lon-z triplets) and can also do some elementary interpolating and smoothing.

By keeping much of the data in ASCII format, reformatting and processing tasks can be handled through the UNIX utilities *awk, grep, sed,* etc. and easily made available to graphics utilities favored by individual users.

The format for storing 3- and 4-D gridded datasets is based on the XDR platform-independent data representation implemented in the netCDF libraries available through UCAR. This format was chosen for a number of reasons: it was designed to handle gridded data, it is becoming a standard in the scientific community, many utilities which handle netCDF files already exist, and it permits transporting data across computer architectures.

The following sections explain the usage of these utilities and detail some of the algorithms used in their implementation.

1. Gridding Hydrographic Properties in 3-D: *grid3d*

Input: *HydroBase* format files, optional sigma series definitions, std depths

Output: cdf file (binary netCDF format)

Usage: **grid3d** *filename_root(s)* **-B***west/east/south/north* **-I***x_incr/y_incr*
    -P*list_of_properties* **-C***cdf_outfile_name* [**-N**] [**-G**] [**-S***ref_id/sigma_series_file*]
    [**-Z***std_depth_file*] [**-T***time_bin_file*] [**-W***window[/w_incr]*] [**-D***dirname*] [**-E***file_extent*]

filename_root(s): are the list of *HydroBase* files to be searched. You can specify the entire path for each file or just the rootname and use the -D and -E options to specify directory and file extent. The filenames *must* be the first arguments after the grid3d command.

-B : specifies grid bounds

-C : name of netCDF output file.

-I : specifies grid increment in degrees; *ex:* -I0.5
    OR specify separate x,y increments with a slash to delimit xincr/yincr;
    ex: -I2.0/0.5

-P : list of property IDs to incorporate;
    ex: -Ppr/th/sa/ox/ht
        -P (by itself) produces a list of available property IDs

[**-S**] : ref_id = 0,1,2,3 or 4. Specify a separate file for each reference level.
    Each line in file contains sigmin, sigmax, incr (sigmin and sigmax are
    inclusive in generating series). Values *MUST* be monotonically *increasing*.
    -S by itself lists the default sigma series.

[**-Z**] : name of file containing list of standard depth levels to use in output matrix.
    (values *must* be monotonically increasing.)
    -Z by itself lists the default standard depth series.

[**-T**] : name of file defining time bins to grid multiple matrices.

[**-N**] : suppress inclusion of observation counts.

[**-G**] : report vertical data gaps in the output profiles to the screen (stderr).

[**-D**] : specifies directory for input data files (default is current directory)
    ex: -D../data

[**-E**] : specifies input file extent (default is no extent)
    ex: -E.dat

[**-W**] : pressure window (db) around each observation for computing gradient properties (buoyancy,
    vorticity...). This window is subdivided into a pressure series for approximating the t and s
    gradients. The optional *w_incr* gives the size of the increment (db) of this pressure series. You
    gain nothing, but expend more cpu cycles, by specifying an increment smaller than the vertical
    resolution of the input data. The value of *window* should specify <u>half</u> the total pressure
    window. The default value for *window* is 10. If no *w_incr* value is specified, *w_incr* is set to the
    same value as *window* so that 3 points will determine the gradient .
    ex: -W100/10 means a gradient window of 200 db (100 above, 100 below) will be used to compute the
    density gradient. Within that window, the observations will be interpolated onto 10 db intervals to
    compute the t and s gradients used in computing the density gradient.

Grid3d computes an average profile at each node on a latitutude/longitude grid from randomly spaced stations and outputs a netCDF file of hydrographic properties gridded as a function of lat, lon, and depth. The depth dimension consists of a series of standard depths which have default values, but which can be optionally specified using **-Z**. The properties to be profiled, which are chosen from a list of options, are averaged on *density* (sigma) surfaces by interpolating the observations onto a series of density levels and computing a mean for all observations within a grid square. The user specifies the isopycnal (sigma) surfaces on which the averaging with the **-S** option (see below). Unless the **-N** option is specified, the number of observations used to compute each average is output together with the averaged value. With the **-T** option, multiple 3-D matrices will be created, one representing all the data, while each of the others represents a user-specified range of years. When this option is not specified, a single matrix incorporating all data will be generated. By presorting the station data with the *extract* module (e.g. by month, depth levels, location...), you can selectively filter the dataset which you input to *grid3d*.

Isopycnal Averaging at the Sea Surface:

Near the sea surface, temporal variations in wind stress, heat, and freshwater fluxes considerably complicate isopycnal averaging. *Grid3d* uses a mixed-layer approach in determining the average properties at the sea surface. A mixed layer is defined for each observed station as the depth range where the density is within 0.1 sigma-0 units of the value at the sea surface (see figure 9). The mean of all observed sea surface density classes, equally weighted, determines which density class will ultimately characterize a gridnode. Each mixed layer class, thus defined, has an associated thickness (depth) which is the average depth of all the observed mixed layers that fall into the same class. Measured properties (pressure is, of course, an exception) are assumed to be uniformly mixed throughout the mixed layer. Thus a single value for each property is determined for an observed mixed layer by integrating the observed property values with respect to the pressure interval over which they were observed and dividing by the total pressure range to yield a property value per unit volume. This value, determined for each station, is summed with other stations of the same surface density class to produce a mean property value which is applied evenly throughout the mixed layer chosen to represent the gridnode. Conversely, values for pressure and derived properties (e.g. sigma, dynamic height...) are computed for the top, bottom, and at 100 meter intervals throughout the mixed layer from pressure, temperature, salinity triplets at these levels. The values of properties at depth levels beneath the mixed layer are determined from the data which was isopycnally averaged on the specified sigma series. When interpolating the isopycnally averaged data onto the depth levels of the 3-D grid,the algorithm checks for and uses only the sigma surfaces which are deeper and denser than the bottom of the mixed layer to avoid creating any static instabilities at the base of the mixed layer. In other words, the depth levels of the output grid which fall within the mixed layer depth will have a uniform value for each property, while the property values associated with gridded depth levels deeper than the bottom of the mixed layer will be determined by interpolating the values isopycnally averaged in the density series.

Choosing Sigma Surfaces :

Although a default density series on which the properties are averaged is supplied, the user can (and should!) specify a sigma (density) series with the **-S** options. Optimum results will be obtained by customizing the sigma series for a particular part of the ocean. The goal is to specify sigma values which span the entire water column and adequately characterize the property profiles. Because the isopycnally averaged properties will be interpolated back onto standard depths, a closely spaced set of sigma surfaces will impart greater accuracy to the determination of depth profiled properties. However, a large number of sigma surfaces will slow the speed of the computation significantly. The sigma series therefore, should be specified to balance speed and accuracy, and should reflect the vertical density gradients for a particular area.

The sigma series is comprised of five different reference pressures (0, 1000, 2000, 3000, 4000 db); and values for each reference pressure should be specified separately using **-S**<*ref_id*>, where *ref_id* is 0, 1, 2, 3, or 4. Because sigma is not a linear function of T, S, and P, the transition points between sigmas referenced

Sigma-0

Grid3d defines a mixed layer at each station as sigma$_{ss}$ + 0.1, maintains a list of all mixed layer classes encountered at each grid point, and averages together properties of stations which belong to the same class.

The mean density of all (mixed layer) bins determines which mixed layer class will characterize an individual grid point.

Depth levels for the averaged grid node which fall between 0 and the depth of the mixed layer are assigned uniform property values determined from the mixed layer bin. Property values at deeper levels are determined from the isopycnally averaged density series starting with the first density level in which the averaged depth exceeds the depth at the bottom of the mixed layer...

sea surface
(average T , S, O2 ....)

depth of mixed layer (~ 60 meters)

Sigma-0

500 m

Sigma-1000

1500 m

Figure 9: Density profiles of upper 100 meters of stations within 1° square area illustrates seasonal variation of density at sea surface (SS). *HydroBase* uses a mixed layer approach to estimate the average property values at and near the SS: schematic bins depict the average depth and sigma distribution of "mixed layer classes" which would be defined for this grid node. The highlighted bin constitutes the mixed layer class which would represent this node and determine its property values from the sea surface to ~60 meters.

to consecutive pressure levels (e.g. the transition from sigma-0 to sigma-1000 values) are uniquely determined at each gridnode. Although properties will be averaged on every surface specified in the sigma series, the averaged depth values determine which sigma surfaces will ultimately be used to compute the property profiles in the depth ranges associated with each reference pressure (0-500 m for sigma-0, 500-1500 m for sigma-1000, 1500-2500m for sigma-2000, etc...). For example, only sigma-1000 levels whose <u>averaged</u> depth is greater than 500 meters and less than 1500 meters will be used in determining the profile at standard depths between 500 and 1500 meters. This somewhat eases the task of choosing sigma surfaces for the isopycnal averaging -- since a single set of sigmas will suffice for large geographic areas -- while it also avoids problems of jumps and instabilities at reference level boundaries.

Choose your sigma levels by determining the range of sigma-0 values which occur in the depth range 0-500 meters. You can look in atlases to find this information, or extract a few stations from the region and use propcalc to compute sigma values for each station:

<div align="center">propcalc sample.dat -Ppr/de/te/sa/s0/s1/s2/s3/s4 > sample.prop</div>

Do the same for sigma-1000 (depths between 500-1500 meters), sigma-2000 (1500-2500 meters), sigma-3000 (2500-3500 meters), sigma-4000 (>3500 meters). You should aim to overspecify, rather than underspecify, so you don't have large vertical gaps between consecutive isopycnals. Examples of sigma files can be found in the ./lists subdirectory of this software distribution.

<u>Vertical</u> <u>Data</u> <u>Gaps</u>:

Interpolating between vertically spaced observations remains one of the great problems associated with hydrographic data. Poor sampling strategy, failure of the sampling equipment, and errors in analysis represent a few of the many possible sources of inadequate vertical resolution. Where property gradients are large and varying relative to the vertical spacing, interpolation may produce invalid estimates of the properties between observations. The non-linearity of the equation of state will then introduce significant error to derived properties such as density. No single method, linear or splines of many varieties, produces an adequate solution to missing data in all cases. Under the assumption that *no* data is better than *bad* data, *HydroBase* checks for and does *not* interpolate over vertical datagaps. In the thermocline (upper 1000 meters) a datagap is defined as >200 meter separation between observations, while below that level, the limit is relaxed to >600 meters. This check applies to both incoming data--the observations which are interpolated onto the sigma series--and outgoing data--when interpolating the isopycnally averaged profiles back onto standard depths. This results in average profiles with some holes (which are filled with missing data flags in the netcdf files), but also prevents anomalous blips in the property fields. The -G option will enable a report of datagaps which occur in the output profiles. A datagap occurs when a standard depth level falls between two density surfaces whose average depths differ by 200 m in the thermocline (or 600 m beneath the thermocline). This could indicate that datagaps in the original station profiles prevented any information from being averaged in this depth and density regime. The software does attempt to distinguish between datagaps and a pycnostad in which a very thick density layer could result in a distance > 200 m between density levels. Vertical interpolation is performed over a pycnostad.

<u>Computing</u> <u>Averages</u> <u>of</u> <u>a</u> <u>Property</u> <u>vs.</u> <u>Computing</u> <u>a</u> <u>Property</u> <u>from</u> <u>Averages</u>:

Every user of this package should understand that computing properties such as sigma, dynamic height, buoyancy, etc. at each station and finding the average value of that property is not necessarily the same as computing these properties from averaged t, s and p profiles. This module was intentionally constructed to permit this to be done either way. Experience has shown that in general, generating average profiles of observed properties (pr, te, sa, ox...) while computing the derived properties from those averages produces sensible values.

Grid3d can be run consecutively to grid average profiles from observed properties, then to construct average profiles of derived properties computed from the first set of average profiles by using *cdf2asc* to convert the cdf file into a *HydroBase* format file. Although perhaps confusing, this concept has valuable applications when your goal is to construct 3-D matrices of derived properties.

Other Notes:

Grid3d is a memory-and-compute-intensive program which can take considerable time to run depending on the memory resources and speed of your cpu. However, *gridsurf,* which reads cdf format files, works very quickly from the netcdf files, so once *grid3d* has been run, subsequent steps are relatively speedy. The code dynamically allocates and frees memory as it is needed, but you may find it necessary to subdivide a large geographic area and produce multiple .cdf files to avoid running out of program memory. **It is wise to run *grid3d* separately for each 10° Marsen Square file** (see the script file *dogrid3d* in the ./extras subdirectory of this software distribution.) Multiple netcdf files can be input to *gridsurf.*

Comparisons of property maps on density surfaces which are generated by *gridsurf* from *grid3d* output and from the original ascii data files are virtually identical. We deemed this a good test of the *grid3d* averaging algorithms.

examples:

grid3d -P

produces a list of available properties.

grid3d -S

lists the default sigma series.

grid3d -Z

lists the default standard depths.

grid3d 7001 7002 7003 7004 7005 -B-60/-10/0/10 -I.5 -Ppr/th/sa/ox -Cequator.cdf -E.dat -S0tropics.sig0 -S1tropics.sig1 -S2tropics.sig2 -S3tropics.sig3 -Ttime.bins -Zstddepth.list

will use all stations in the files 7001.dat 7002.dat ... etc. to create a gridded dataset of pressure, potential temperature, salinity, oxygen. The lat/lon grid bounds will be 60°-10°W, 0-10N with half-degree spacing; the depth values listed in the file *stddepth.list* will constitute the depth dimension. The averaging will be done on the density surfaces specified in the files *tropics.sig0, tropics.sig1,...*etc; default sigma-4 values will be used. The output file will be called *equator.cdf.* In addition to a matrix including all stations, two other matrices will grid the averages for the ranges of years specified in *time.bins.*

The file *tropics.sig0* contains:
18.0 25.0 1
25.1 27.0 .1
27.02 28.2 .02

The file *time.bins* contains:
1950 1969
1970 1999

The file *stddepths.list* contains:
0 10 20 30 50 75 100 125 150 200 250 300 400 500 600 700 800 900 1000 1100
1200 1300 1400 1500 1750 2000 2200 2400 2600 2800 3000 3200 3400 3600 3800
4000 4200 4400 4600 4800 5000 5200 5400 5600 5800 6000 6200 6400 6600 6800
7000 7500 8000 8500 9000 -99

NOTE: -99 holds a place for the bottom observation at each gridnode. The last element in the standard depth array will always be replaced with the deepest observation.

2. Gridding on a 2D Surface: *gridsurf*

Input: *HydroBase* station files or cdf_files

Output: ASCII text file for each surface specified.

Usage: **gridsurf** *filename_root(s)* [-C*time_bin_index*] -B/*west/east/south/north*

      -I*increment* -P*list_of_properties* [-S*surface_definition_file*] [-W*window*[/*w_incr*]] [-D*dirname*]

      [-E*file_extent*]

filename_root(s): are the list of *HydroBase* files to be searched. You can specify the entire path for each file or just the rootname and use the -D and -E options to specify directory and file extent. The filenames *must* be the first arguments after the command.

  -B : specifies grid bounds

  -I : specifies grid increment in degrees; *ex:* -I0.5

  -P : list of properties to project onto surface;

      ex: -Ppr/th/sa/ox/ht

        -P (by itself) produces a list of available properties

[**-C**] : input files are cdf format files. (default is *HydroBase* format)

      optionally specify time_bin index (if none specified, index 0 will be used)

[**-S**] : file containing surface definitions.If this is *not* specified these values are expected

      to come from the standard input device.

[**-D**] : specifies directory for input files (default is current directory)

      ex: -D../data/

[**-E**] : specifies input_file extent (default is no extent)

      ex: -E.dat

[**-W**] : pressure window (db) around each observation for computing gradient properties (buoyancy, pot.vorticity). This window is subdivided into a pressure series for approximating the t and s gradients. The optional parameter *w_incr* gives the size of the increment (db) of this pressure series. You gain nothing, but expend more cpu cycles, by specifying an increment smaller than the vertical resolution of the input data. The value of *window* specifies half the total pressure window. The default value for *window* is 10. If no *w_incr* value is specified, *w_incr* is set to the same value as *window*. ex: -W100/10 means a gradient window of 200 db (100 above, 100 below) will be used to compute the density gradient. Within that window, the observations will be interpolated onto 10 db intervals to compute the t and s gradients used in computing the density gradient.

Gridsurf projects hydrographic properties onto one or more surfaces and computes mean and standard deviation values of those properties for each node in a grid. The grid spacing and bounds are specified by the user. All points within a grid square are used to compute the mean and are weighted equally; squares do not overlap. The *centers* of each square form the nodes of the matrix representing a surface.

The surfaces may be defined by some value of any property supported by *HydroBase* (depth, pressure, temperature, density, etc.) The properties at each station are linearly interpolated onto the surface; and the averaging, which produces the grid, is done on that surface. That is, if the surface is a 100 m depth surface, the interpolated properties will be averaged on that depth surface. *WE STRONGLY RECOMMEND* that properties be averaged on *density* surfaces to avoid distorting the t-s property relationships, especially in frontal zones. If the surface being gridded with *gridsurf* is of a type other than density, the data should first be gridded (averaged) with the program *grid3d*, and that product used as input to *gridsurf*. The input data may be randomly spaced stations or pre-gridded data: *gridsurf* accepts *HydroBase* station files or cdf files produced by *grid3d*.

For each surface, an output file of ASCII text is generated containing:

nrows, ncols, gridspacing, gridbounds, nprops	(1st line)
list_of_properties	(2nd line)
lat lon p_1 p_1dev n_1 p_2 p_2dev n_2 ...	(1 line for each gridpt)
where for each property: p = mean	
$pdev$ = std dev	
n = # of obs	

Some specific details regarding the inner workings of *gridsurf*:

a. *Gridsurf* uses the first (shallowest) occurrence of the value defining the surface. This is important only when the property defining the surface is not monotonic with depth.

b. If the surface is a density surface and is absent from a particular station, *gridsurf* checks to see if the surface is outcropping at that station. If the observed density at the sea-surface is greater than the density for this surface, *gridsurf* accounts for the outcropping by pretending the surface occurred just above the sea surface at a depth of -1 meter. This must be done for pressure and depth, but cannot be done for other properties like temperature; so the number of observations which contribute to the mean of these properties must be totalled separately and may in fact differ. *Failure to account for outcropping surfaces would ultimately bias the density surfaces which outcrop seasonally to deeper average depths and pressures than what is observed.*

c. If contiguous observations are missing such that a gap exceeding 200 meters (in the upper 1000 meters) or 600 meters (everywhere else) exists for a property, *gridsurf* will not interpolate over that gap to find the surface. If the surface occurs within that gap, *gridsurf* treats the situation as though the surface does not exist at the station. This strategy largely avoids creating bad values by linearly interpolating property values where the property-depth relation is not linear: in the thermocline, for example.

d. For derived properties (i.e. those computed from measured properties), be aware that there may be a difference between 1) computing the derived property *in situ* and summing those values to find the mean; and 2) computing the derived property from the mean values of the measured properties. This is not a problem with *HydroBase* station data input; the mean is computed by deriving the property *in situ*, summing those values and dividing by the number of stations used. However, with cdf input, *gridsurf* will first check the cdf file to see if a requested derived property is available. In this case, *gridsurf* will use these stored values *whenever that property is referenced.* If the property is *not* available, then *gridsurf* will attempt to compute it from the means of the appropriate measured properties. If any one of these properties is unavailable in the cdf file, *gridsurf* will not be able to compute the derived property and will print a message on the *stderr* device. Although this is not a fatal error, the user should be aware that the resulting output may differ from the desired product.

e. For cdf input, no standard deviations are computed.

f. If the input is a cdf file and the counts used to compute each mean value were included during grid3d (i.e. **-N** option not specified), then that information will be carried through by *gridsurf* and output to the surface files. If no counts were included *or* if the property is not stored in the cdf file, but can be computed from the stored properties, then each gridded output point is treated as if a single observation determined its mean value.

24

examples:

gridsurf -P
will cause a list of properties to be printed on the stderr device.

gridsurf 7105 7106 7205 7206 -B-70/-55/10/30 -I.5 -Ssurf.lis -Ppr/th/sa/ox -D../data -E.dat
will project pressure, potential temp, salt, and oxygen onto the surfaces listed in *surf.lis* for the area spanning 70-55 W, 10-30 N. The output files are specified in *surf.lis;* the output grid has half-degree spacing between nodes.

contents of *surf.lis*: (*property_id   value   output_file_name*)
s0 26.5 sig2650_hdeg.grid
s1 31.85 sig3185_hdeg.grid
s_ 32.35 1500 sig3235_hdeg.grid
s2 36.95 sig3695_hdeg.grid
s4 45.90 sig4590_hdeg.grid
end

gridsurf natl.cdf -D/home/ruth/cdf -C -B-85/0/0/65 -I1.0 -Ssurf.list -Pht/va/pr
will project dynamic height, specific volume anomaly, and pressure onto the surfaces listed in surf.list for the area spanning 0°-85° W, 0°-65° N. The output grid has 1° spacing in both lat and lon directions. The binary cdf file created by *grid3d* is used as input.

3. Smoothing a Gridded Surface: *smoothsurf*

Input: ASCII file created by *gridsurf*

Output: List of lon/lat pairs

Usage: **smoothsurf -I***input_filename* **-O***output_filename* **-P***list_of_properties* **-M***minobs* **-R***search_radius* [**-N**] [**-S**] [**-B***blanking_info_file*]

  -I : name of input file (produced by *gridsurf*)

  -O: name of output file

  -P: specifies a list of property ids to smooth;

     ex: **-P**pr/th/sa/ox

  -M: minimum number of observations desired per gridnode

  -R: search radius (number of gridpoints)

 [**-N**]: include number of observations and radius of smoothing in output file

 [**-S**]: generate standard deviation information

 [**-B**]: specifies name of file containing list of gridpoints to be automatically blanked.

    Smoothsurf divides by individual properties the file produced by *gridsurf* and creates files containing lon, lat, z triplets. It can additionally perform some simple interpolation and smoothing by using the **-M** and **-R** options. If the total number of observations is fewer than *minobs*, a value specified with **-M** option, each point lying one node farther away is added, weighted not only by its number of observations, but also by a distance factor $1/2d$ (where d is the number of gridpoints away from the central node). The user specifies how far out (radius of influence) to search to find *minobs* number of observations.

    The program outputs up to 4 files for each property specified:

file type	contents	file_name
1) property info:	lon, lat, av_value, [radius, n]	*<root><property_id>*.xyz
2) blank gridpts:	lon, lat	*<root><property_id>*. blk
3) stddev info:	lon, lat, stddev, [radius, n]	*<root><property_id>*_var.xyz
4) stddev blanks:	lon, lat	*<root><property_id>*_var.blk

where *root* is supplied by the **-O** option and *property_id* is the same as the property_list_ids in the -P option.

    The output files for blank gridpoints (file types 2 and 4 above) contain a list of lon/lat pairs representing the gridnodes at which there were zero observations after smoothing. This information is intended to be used in graphics routines which permit the user to specify areas to mask prior to contouring.

    The **-B** option permits you to supply a file containing the lon, lat of gridnodes to be *automatically* blanked. It provides extra control in masking out bathymetry, coastlines, etc... Before computing each new gridpoint, *smoothsurf* checks this list of gridnodes. If the gridpoint is on the list, then it is immediately listed in the blanking file(s) without any computation performed for that node.

The -M and -R options together control the radius of influence (actually a square) used in smoothing each gridpoint. The algorithm implemented in this module permits the smoothing radius to vary based on the local observation density. Because smoothing dampens the intensity of the property gradients, especially in frontal zones such as the Gulf Stream, it is desirable to apply the least amount of smoothing to a property field while yet providing statistically meaningful mean values whenever possible. This approach presumes that if a gridpoint is defined by an ample number of observations, it need not be additionally smoothed. Conversely, where only a few or no observations define a gridpoint, information from neighboring nodes should be incorporated to define the mean property values. This is especially applicable to hydrographic data because station density is not random. High sampling densities are clustered near-shore and around frontal zones such as the Gulf Stream; low station densities predominate in the gyre interiors. According to the criteria above, smoothing will tend to occur in contiguous areas and will generally coincide with quiescent zones; regions characterized by sharp property gradients will require little or no smoothing.

When the gridded data files include the number of observations which contributed to each mean value, we use *minobs* = 10 and *radius of influence* = 2 for a 1-degree resolution of the North Atlantic. Statistically, this number of observations guarantees us 90% confidence that a computed mean value will fall within one half of a standard deviation from the true mean for a normal distribution of values. These parameters can and should be adjusted to a particular application and the judgement of the user. They are designed to permit flexibility: e.g. a very large *minobs* value will force the smoothing to be uniformly applied according to the *radius of influence* (-R value) specified; using *minobs*=1 will interpolate only at points where there are no data.

When the *gridsurf* files do *not* have information on the number of observations (or if it is set to 1 because the property was computed from an averaged p,t,s), then *minobs* =1 and *radius of influence* = 2 will interpolate over holes in the data on this surface. Setting *minobs* > 1 will result in uniform smoothing according to the radius specified. Using *radius of influence* = 0 will result in no smoothing, however, all properties in the *gridsurf* .grid file will be sorted into individual .xyz property files.

examples:
smoothsurf -Isig4590.grid -Osig4590 -Ppr/de/th/sa/ox -M10 -R2

will divide the file sig4590.grid into various property files (sig4590pr.xyz, etc) and smooth the data to get 10 observations contributing to each gridpoint. It will, however, only use information within 2 gridnodes of any particular point.

smoothsurf -Isig4590.grid -Osig4590 -Ppr/de/th/sa/ox -M1 -R2

will divide the file sig4590.grid into various property files (sig4590pr.xyz, etc) and interpolate only at gridpoints where there is no data (min obs = 1). Again it will only use information within 2 gridnodes of any particular point.

smoothsurf -Isig4590.grid -Osig4590 -Ppr/de/th/sa/ox -R0 -M1

will divide the file sig4590.grid into various property files (sig4590pr.xyz, etc), but do no interpolation or smoothing of any gridnodes.

4. Blanking Gridnodes: *preblank, findblanks*

Input: ASCII file of gridded data on a surface of the type created by *gridsurf*

Output: list of lon/lat pairs representing blank squares

Usage: **preblank** -I*input_filename* -O*output_filename* -P*property_id* -R*search_radius*

-I : specifies the input filename;

-O : speciifies the output filename;

-P : 2-character property ID;

-R : maximum radius away (in gridpoints) to search for observations;

Preblank determines which gridnodes in a matrix of points representing a surface will contain no data after smoothing the specified property. Its purpose is to create a blanking file representing coastlines, out-cropping bathymetry, etc. for use in smoothing and contouring. The output file contains a list of lat/lon pairs corresponding to gridnodes which contain no data. The algorithm used prevents oversmoothing at the edges by evaluating the distribution of points within the radius of influence surrounding each gridnode. At least one point must be present on both the left *AND* right side of the gridnode in order to incorporate both left and right points into the central point; this criterion is similarly applied to the top and bottom sides.

examples:

preblank -Isig2650_hdeg.grid -Opr2650.blank -Ppr -R3

will use the file *sig2650_hdeg.grid* (created by *gridsurf*) and evaluate the distribution of pressure observations to determine which gridnodes will contain zero observations after smoothing with a search radius of 3 points . The list of lon/lat pairs will be output to *pr2650.blank.*

findblanks

Input: ASCII file of xyz values (similar to what smoothsurf produces)

Output: list of lon/lat pairs representing blank squares

Usage: **findblanks** *filename* -B*west/east/south/north* -L*xincr/yincr* > *outfilename*

-B : specifies the geographical boundaries of the grid;

-I : specifies the lon/ lat grid increments

Findblanks reads a file of gridded xyz values (lon, lat, z) and identifies the gridpoints which contain no data. It writes a list of lon/lat values to the stdout device. This permits you to create a blanking file specific to this grid for a graphics application.

example:

findblanks sig3690pr.xyz -B-80/-60/20/45 -I.5 > sig3690.sargasso.blank

5. Searching for and Extracting Stations from HydroBase: *extract*

Input: *HydroBase* station files

Output: *HydroBase* station (to *stdout* device)

Usage: **exract** *filename_root(s)* [-D*dirname*] [-E*extent*] -T*list_of_extraction_types*
[-O*outcast_file*] > *outfile*

filename_root(s): are the list of *HydroBase* files to be searched. You can specify the entire path
for each file or just the rootname and use the -D and -E options to specify directory and file_extent.
Filenames *must* be the first arguments after the command.

[-D] : specifies directory in which the input files are stored (default is ./)
    ex: -D../data/

[-E] : specifies file extent for the input files (default is no extent)
    ex: -E.dat

[-O] : specifies file to store stations that do *not* meet criteria.

-T : specifies type of extraction and criteria

    c = cruise (nodc cruise code)   -T**c**/code1/code2/code3...
    d = station depth (deepest observation, in meters)   -T**d**/depthmin/depthmax
        will extract *entire* station if deepest observation is between specified min/max
    g = geographic position (W and S are negative)   -T**g**/minlon/maxlon/minlat/maxlat
    ld = extracts only observations between specified depth levels   -T**ld**/mindepth/maxdepth
    lp = extracts only observations between specified pressure levels -T**lp**/min_pr/max_pr
    m = month (1=Jan; 12=Dec)   -T**m**/month1/month2/month3...
    n = nation (nodc country code)   -T**n**/code1/code2/code3...
    s = ship (nodc ship code)   -T**s**/ship1/ship2/ship3...
    y = year   -T**y**/minyear/maxyear

Combine any of the above using / to separate the extraction types:
    Tg-80/-10/0/60/**m**1/2/3/12

Use ^ after an extraction type to extract all stations *EXCEPT* the ones which meet the criteria.
    ex: -T**s**^VE/**y**1982/1982
will extract all stations from 1982 except those collected by the ship VE.

Extract will retrieve stations from *HydroBase* according to criteria specified by the user.

examples:

extract 7403 7404 7405 7503 7504 7505 -D../data -E.dat -Tg-55/-35/40/55 > grandbanks.dat
will extract from the input files (7403.dat,7404.dat,...etc.) located in the directory ../data, the stations which fall
in the area 55-35°W, 40-55°N and direct the output to a file called grandbanks.dat.

extract 7102 -E.dat -Ty/1955/1960/sCH/AT/d2000/10000 > 7102.igy.dat

will extract from the file 7102.dat in the current directory all stations collected by the Chain and Atlantis
between 1955 and 1960 which were deeper than 2000 meters and direct the output to a file
called 7102.igy.dat

extract 7102 -D../data -E.dat -Tm1 -O7102.rem > 7102.jan

will extract from the file ../data/7102.data all stations collected in January, output them to a file called 7102.jan
and output the rejected stations to a file called 7102.rem

6. Listing information from a CDF file: *cdfinfo   cdf2asc*

Input : netCDF file created by the HydroBase routines
 Output: summary of information to the stdout device
  Usage: **cdfinfo** *cdf_filename*
           cdf2asc *cdf_filename* [-T*time_bin_index*]

   [-T] : selects the matrix # for the appropriate time bin.

These utilities read the cdf files created by *grid3d.*

Cdfinfo will print a summary of the type of information contained in the file.

Cdf2asc will create a version of the actual data values in a format resembling the ascii *HydroBase* station files. The -T option, if specified, selects the matrix associated with the *n*th time bin stored in the file. If not specified, only the first time bin -- which by definition contains all years -- is output. The appropriate time bin index can be found using *cdfinfo*.

7. Reading & Writing HydroBase Files: h*ydro_utils.c*

This file contains a set of functions to facilitate reading and writing HydroBase format files.

int open_hydro_file(char *dir, char *root, char *extent, int print_msg):
    opens an existing hydro file for reading.

int get_station(int file, struct HYDRO_HDR *h_addr, struct HYDRO_DATA *d_ptr) :
    reads a complete station.

int read_hydro_hdr(int file, struct HYDRO_HDR *haddr):
    gets station header info.

void report_status(int status, file *file) :
    reports any errors returned by read_hydro_hdr()

int get_data_scan(int file, double *scan, int n, int *prop_order) :
    reads the next scan in an open hydro file

int get_separator(int file) :
    advances file beyond the station separator

int create_hydro_file(char *fname, int mode):
    creates a new hydro file for writing.

int write_hydro_station(int file, struct HYDRO_HDR h, struct HYDRO_DATA data) :
    writes an entire station to an open hydro file

int write_hydro_hdr(int file, struct HYDRO_HDR h) :
    writes a header to an open hydro file

int write_hydro_scan(int file, double *scan, int n, int *prop_order) :
    writes an individual data scan

int write_separator(int file) :
    writes a separator line

int ms10(float lat, lon, int *ms1_ptr) :
    Computes the 10-degree & 1-degree Marsden Square # for a lat,lon pair.

int available(enum property p, struct HYDRO_HDR hdr) :
    Returns a 1 if the specified property is available for a station.

8. Reading and Writing cdf_files : *hydro_cdf.c*

A set of functions to facilitate the creation and reading of netCDF files in conjunction with *HydroBase* software. These functions utilize the UCAR netCDF C-interface routines, which implement XDR.

int cdf_init :	Opens a new CDF file for writing.
int cdf_open :	Open a cdf file for reading.
void cdf_close :	Close a cdf file.
int cdf_define :	Defines all variables and attributes for cdf file.
int write_prop_cdf :	Writes values of a property at a lat/lon node.
int write_prop_count_cdf:	Writes # of obs for a property at a lat/lon node.
int write_std_depths_cdf:	Writes standard depths to de variable.
int write_time_bins_cdf :	Writes min/max values defining time bins.
int write_bottom_depth_cdf:	Writes the bottom depth values to bottom variable.
int read_cdf_hdr :	Reads header info from a cdf file.
int read_cdf_prop :	Reads all values of a property at a lat/lon node.
int read_cdf_prop_count:	Reads # of obs for a property at a lat/lon node.
int read_cdf_depths :	Reads values of depth variable from cdf file.
int read_cdf_bottom_depth :	Reads values of bottom depth from cdf file.
int get_indices :	Converts lat/lon to row/col for a cdf file.
int get_lat_lon :	Converts row/col to lat/lon.
int d2stdlev(depth) :	returns stddepth index associated with depth
double stdlev2depth(index):	returns depth associated with stddepth index.

The default set of standard depths is defined here also:

```
#define NDEF_DEPTHS   41
double def_depth[NDEF_DEPTHS] = {  0,  10,  20,  30,  50,  75, 100, 125, 150, 200,
                  250, 300, 400, 500, 600, 700, 800, 900,1000,1100,
                  1200,1300,1400,1500,1750,2000,2500,3000,3500,4000,
                  4500,5000,5500,6000,6500,7000,7500,8000,8500,9000,
                  -99 };
```

NOTE: The -99 holds a place for the bottom observation

This structure, which is used extensively with the cdf files, is defined in *hydro_cdf.h* :

```
struct CDF_HDR {
    int nx;                 /* Number of columns */
    int ny;                 /* Number of rows */
    int nz;                 /* Number of depth levels */
    int nt;                 /* Number of time bins */
    int nprops;             /* Number of properties */
    int counts_included;    /* 1/0= counts for each property are included or not */
    int node_offset;        /* 0 for node grids, 1 for pixel grids */
    float fill_value;       /* to denote missing data */
    float xmin;             /* Minimum x coordinate */
    float xmax;             /* Maximum x coordinate */
    float ymin;             /* Minimum y coordinate */
    float ymax;             /* Maximum y coordinate */
    float xincr;            /* x increment */
    float yincr;            /* y increment */
    int *tmin;              /* min year for each time bin */
    int *tmax;              /* max year for each time bin */
    char x_units[80];       /* units in x-direction */
    char y_units[80];       /* units in y-direction */
    char z_units[80];       /* units in z-direction */
    char t_units[80];       /* units in time dimension */
    char **prop_id;         /* 2-char mnemonic of various properties */
    char **prop_units;      /* units of various properties */
    char title[80];         /* name of data set */
    char command[320];      /* text of generating command */
};
```

The cdf files contain matrices of hydrographic properties gridded as a function of longitude, latitude, depth, and time. These 4 dimensions are standard to other oceanographic software such as EPIC, hence we adopted them here so that other applications could be used with *HydroBase* files.

Further information on netCDF can be found in the *NetCDF User's Guide* published by UCAR and available by anonymous ftp from unidata.ucar.edu (Internet 128.17.140.3).

9. Computing hydrographic properties: *propcalc.c prop_subs.c phyprops.f   eos80d.f*

In addition to the properties stored explicitly in the database, *HydroBase* routines currently support a small list of computed properties:

     pr : pressure [dbars]
     de : depth [meters]
     te : in situ temperature [degrees C]
     sa : salinity [psu]
     ox : oxygen [ml/liter]
     n2 : nitrite [micromole/kg]
     n3 : nitrate [micromole/kg]
     p4 : phosphate [micromole/kg]
     si : silicate [micromole/kg]
     th : potential temperature: pref= 0. [degrees C]
     ht : dynamic height [dyn. meters (10 m2/s**2)]**
     s0 : potential density: pref = 0. [kg/m3]**
     s1 : potential density: pref = 1000. [kg/m3]**
     s2 : potential density: pref = 2000. [kg/m3]**
     s3 : potential density: pref = 3000. [kg/m3]**
     s4 : potential density: pref = 4000. [kg/m3]**
     s_ : potential density: pref = ?. [kg/m3]**
     bf : buoyancy frequency [1.e^-5 radians/sec]
     pv : potential vorticity [1.e^-12 m^-1 sec^-1]
     sv : specific volume [1.e^8 * m3/kg]**
     va : specific volume anomaly [1.e^8 * m3/kg]**

A set of Fortran functions and subroutines by Fofonoff and Millard (1983) is included in the files *phyprops.f* and *eos80d*.f which implement the UNESCO standards for computation of oceanographic properties using the 1980 equation of state.

<u>Gradient Properties</u>:

    Properties that are derived from the vertical gradients, such as vorticity and buoyancy, are handled in a slightly different manner from the other properties. For temperature, dynamic height, density, and the observed properties, we compute a value at each observation level then interpolate this property profile to find the value at the desired surface. The value of a derivative property on a surface, however, is computed from the appropriate vertical gradients (temperature and salinity usually) surrounding the desired surface, and not by creating a profile of the derivative property at each observation level.

Propcalc

Input: *HydroBase* station file
Output: *HydroBase* station file
Usage: **propcalc** *filename_root(s)* **-P***list_of_properties* [**-D***dirname*] [**-W***window*[*/w_incr*]] [**-E***file_extent*]
> *outfile*

filename_root(s): are the list of *HydroBase* files to be searched. You can specify the entire path for each file or just the rootname and use the -D and -E options to specify directory and file extent. The filenames *must* be the first arguments after the command.

    -P : list of properties to project onto surface;
      ex: -Ppr/th/sa/ox/ht
        -P (by itself) produces a list of available properties
    [**-D**] : specifies directory for input files (default is current directory)
      ex: -D../data/
    [**-E**] : specifies input_file extent (default is no extent)
      ex: -E.dat
    [**-W**] : pressure window (db) around each observation for computing gradient properties (buoyancy, pot.vorticity). This window is subdivided into a pressure series for approximating the t and s gradients. The optional parameter *w_incr* gives the size of the increment (db) of this pressure series. You gain nothing by specifying an increment smaller than the vertical resolution of the input data. The value of *window* specifies <u>half</u> the total pressure window. The default value for *window* is 10. If no *w_incr* value is specified, *w_incr* is set to the same value as *window*.
    ex: **-W100/10** means a gradient window of 200 db (100 above, 100 below) will be used to compute the density gradient. Within that window, the observations will be interpolated onto 10 db intervals to compute the t and s gradients used in computing the density gradient.

Propcalc computes hydrographic properties at each level in a station and outputs to stdout a *HydroBase* station file which contains all the properties specified with the **-P** option.

examples:
propcalc 7203 -D/d1/data -E.dat -Ppr/te/th/sa/s3/s4 > 7203.sta
will read each station in the HydroBase file /d1/data/7203.dat and output a file in which each station lists the properties pressure, temperature, potential temp, salinity, sigma-3, and sigma-4. Output is redirected from the screen to a file 7203.sta.

Adding a Property to *HydroBase*:

1) in *hydrobase.h*, append a 2-character identifier appropriate for the new property to the *enum property* list; increment the value of the constant MAXPROP.

2) in *prop_subs.c*, append a 2-character mnemonic identifier to the array **prop_mne[]**, a descriptor to the array **prop_descrip[]**, and the units descriptor to the array **prop_units[]**. Also add appropriate values to the **field_width[]** and **field_precis[]** arrays describing the width.precision format specification to be used in printing values of the property. The order is important! Be certain to maintain the one-to-one correspondence between each of the variables in steps 1 and 2.

3) add a function which computes the property to *prop_subs.c*

4) in *prop_subs.c*: in the function **get_prop_indx()**, add an appropriate case to the switch statement;

5) in *grid3d.c:*

      search for the string   **!**!** Special cases**

      to find all the places where an appropriate case should be added:

        1 in function **main()**

        1 in function **mixed_layer_vals()**

6) in *gridsurf.c:*

      search for the string   **!**!** Special cases**

      to find all the places where an appropriate case should be added :

        1 in function **get_hydro_data()**

        1 in function **insertdata()**

        2 in function **get_cdf_data()**:

        - add the appropriate **enum property** case to the **switch** statement to set the **compute_flag** correctly;

        - add an appropriate case to the **switch** statement to compute the property .

7) in *propcalc.c:*

      search for the string   **!**!** Special cases**

      to find all the places where an appropriate case should be added :

        1 in function **get_hydro_data()**

8) **re-make** the entire *HydroBase* package .

10. Sorting, Listing and Other Useful Tools: *ms*sort, getpos, surfdiff, timeprop, xy4gmt*

Sorting Modules:

These modules : *ms10sort, ms5sort, ms2_5sort,* and *ms1sort* will sort files by 10°, 5°, 2.5°, and 1° squares respectively.

Input: *HydroBase* station files

Output: *HydroBase* station files

Usage: **ms*sort** *list_of_filename_root(s)* [**-D***dir*] [**-E***extent*][**-O***outpath*] **-N***new_extent*
     [**-A**] [**-T**] > *logfile*

[**-D**] : specifies directory of input files (default is *./*)   *ex:* -D../data/

[**-E**] : specifies input file extent (default is no extent)   *ex:* -E.dat

[**-O**] : specifies directory of output files (default is *./*)   *ex:* -O../natl/

-**N** : specifies new file extent (default is no extent)   *ex:* -N.dat

[**-A**] : append to existing files (default is do not alter an existing file.)

[**-T**] : truncate existing files (default is do not alter an existing file.)

These modules will read and sort any number of files, but only 50 output files can be simultaneously open. A new file will be opened each time a station is encountered which does not fit into a currently open output file. After the maximum number of files has been opened, such stations will be placed into a scratch file named *msextra.dat.* A summary of station counts will be displayed on the stdout device at the end of the sort. The -**A** and -**T** options permit existing files to be added to or truncated. If neither option is specified, the default (noclobber) will prevent existing files from being overwritten or appended to.

The naming convention for output files is   *ms10_##.extent*
     where: *ms10* is the 4-digit 10° Marsden Square designation,
         _## depends on the dimension of the sorted area, and
         .extent is specified at run time

Dimension of sorted area (# per 10* MSq)	:	*values of _##*
5° files (four)	:	_0, _1, _2, or _3
2.5° files (sixteen)	:	_0h, _1h, ... _9h, _Ah, _Bh, _Ch, _Dh, _Eh, _Fh
1° files (one hundred)	:	_00 .. _99

Listing Station Positions

Getpos will generate a list of longitude/latitude pairs for every station in a list of files. This is particularly useful for making maps of station locations.

Input: *HydroBase* station files

Output: ascii list of longitude latitude pairs

Usage: **getpos** *lst_of_filenames* [**-D***dirname*] [**-E***extent*] [**-B***minlon/maxlon/minlat/maxlat*]

[**-D**] : specifies directory of input files (default is *./*)   *ex*: -D../data/

[**-E**] : specifies input_file extent (default is no extent)   *ex*: -E.dat

[**-B**] : specifies optional boundaries.   *ex*: -B-90/0/0/65

Referencing One Surface to Another

Surfdiff takes two files containing lon, lat, z triplets and produces lon, lat, Δz (where $\Delta z = z2 - z1$), and is useful for referencing one surface to another. Triplets are output only where both files have a datapoint.

Input : 2 xyz files containing lon,lat,z triplets

Output: 1 xyz file

Usage: **surfdiff** *file1 file2* **-B***west/east/south/north* **-I***deltax/deltay* > *output_file*

-I : specifies gridspacing   *ex*: -I.5/1.0

-B : specifies boundaries.   *ex*: -B-90/0/0/65

Projecting Properties onto a Surface (non-gridded) with Time Information : *timeprop*

Timeprop projects hydrographic properties at each station onto one or more surfaces and outputs the values of those properties together with year, month, lat and lon for each station in which the surface occurs. The surfaces may be defined by some value of any property supported by *HydroBase* (depth, pressure, temperature, density, etc...)

Input: *HydroBase* station files

Output: ascii list of year, month, lat, lon, property values

Usage: **timeprop** *list_of_filenames* [-D*dirname*] [-E*extent*] -P*list_of_properties* [-B*minlon/maxlon/minlat/maxlat*] [-S*surf_definition_file*] [-W*window[/w_incr]*]

-**P** : list of properties to project onto surface   *ex*: -Ppr/th/sa/ox/ht
  -P (by itself) produces a list of available properties

[-**D**] : specifies directory of input files (default is ./)   *ex*: -D../data/

[-**E**] : specifies input_file extent (default is no extent)   *ex*: -E.dat

[-**B**] : specifies optional boundaries.   *ex*: -B-90/0/0/65

[-**S**] : file containing surface definitions. If this is not specified, you will be prompted to define the surfaces interactively.

[-**W**] : pressure window (db) around each observation for computing gradient properties (buoyancy, pot.vorticity). The value of *window* specifies half the total pressure window. This window is subdivided into a pressure series for approximating the t and s gradients. The optional parameter *w_incr* gives the size of the increment (db) of this pressure series. You gain nothing by specifying an increment smaller than the vertical resolution of the input data. The default value for *window* is 10. If no *w_incr* value is specified, *w_incr* is set to the same value as *window*.

Extracting X-Y pairs for property-property plots: *xy4gmt*

Input: *HydroBase* station files

Output: ascii list of x and y property values

Usage: **xy4gm** -X*x_property* -Y*y_property* [-I*input_filename*] [-D*mindepth/maxdepth*] [-M]

-**X** : 2-char property mnemonic to specify X

-**Y** : 2-char property mnemonic to specify Y

[-**I**] : optional input file_name . If not specified input will be read from stdin.

[-**D**] : optional depth range for observations

[-**M**] : (optional) separate stations with > symbol

References

Boyer, T. P., and S. Levitus (1994) *Quality control and processing of historical temperature, salinity and oxy gen data*. NOAA Technical Report NESDIS 81. 65 pp.

Fofonoff, Nick, and R. C. Millard (1983) *Algorithms for Computation of Fundamental Properties of Seawater*, UNESCO Technical Papers in Marine Science, Number 44.

Levitus, Sydney (1982) *Climatological Atlas of the World Ocean*, NOAA Professional Paper 13. 173 pp .

Levitus, S., and T. Boyer (1994) World Ocean Atlas 1994, Vol.4: Temperature. NOAA Atlas NESDIS 4. U.S. Government Printing Office, Washington, D.C., 150 pp.

Levitus, S., R. Burgett, and T. Boyer (1994a) World Ocean Atlas 1994, Vol.3: Salinity. NOAA Atlas NESDIS 3. U.S. Government Printing Office, Washington, D.C., 150 pp.

Lozier, M. Susan, M. McCartney and W. B. Owens (1994) *Anomalous anomalies in averaged hydrographic data*, Journal of Physical Oceanography, 24, No. 12, pp 2624-2638.

Lozier, M. Susan, W. B. Owens, and R. G. Curry (1996) *The Climatology of the North Atlantic*, Progress in Oceanography, 36, pp 1-44.

Reid, Joseph L. (1994) On the total geostrophic circulation of the North Atlantic Ocean: Flow patterns, tracers, and transports, *Progress in Oceanography*, 33, No.1, 92 pp.

Smith, W. H. F. and P. Wessel (1990) *Gridding with continuous curvature splines in tension*, Geophysics, 55, pp. 293-305.

Wessel, P. and W. H. F.Smith (1991) *Free software helps map and display data*, Eos Trans. AGU, 72, pp.441, 445-446.

World Ocean Atlas 1994 CD-ROM Data Sets. National Oceanographic Data Center, Ocean Climate Laboratory, Washington, D.C., 9 disks.

Worthington, L. V. (1976) *On the North Atlantic Circulation*. The Johns Hopkins Oceanographic Studies, 6, 110 pp.

APPENDIX A: Structure of the HydroBase Data Files

The boldface type shows a typical station from a *HydroBase* station file.

```
       a  b   c   d    e   f g      h          i        j      k    l  m   n
1☞  26 DN  8  51  1921 11 11   10.267  -40.683 2830   16    4 7104  0
2☞  pr de te sa
3☞      0.0    0.0  27.370  34.790
        25.2   25.0  26.990  35.170
        75.5   75.0  16.760  36.040
       100.6  100.0  14.240  35.500
       125.8  125.0  12.570  35.300
       150.9  150.0  11.060  35.080
       201.3  200.0   9.850  34.930
       302.0  300.0   8.920  34.860
       402.7  400.0   8.420  34.860
       503.5  500.0   7.380  34.780
       604.4  600.0   7.380  34.780
       806.2  800.0   5.730  34.650
      1008.2 1000.0   4.890  34.760
      1210.4 1200.0   4.710  34.880
      1514.0 1500.0   4.260  34.950
      2021.0 2000.0   3.450  34.950
4☞  **
```

1☞ Header line containing information about the station:

a : NODC country code	h : latitude (North positive)
b : NODC ship code	i : longitude (East positive)
c : NODC cruise number	j : echo sounder depth (meters)
d : station number	k : number of observation levels
e : year	l : number of properties
f : month	m : 10° Marsden Sq
g : day	n : 1° Marsden Sq

2☞ 2-character property IDs show type and order of properties in station.

3☞ observed data: 1 line per depth level

4☞ end-of-station indicator

APPENDIX B: Structure of the netCDF Data Files used in HydroBase:

```
netcdf sample {
dimensions:
    lat = 20 ;
    lon = 20 ;
    de = 41 ;
    time = 3 ;

variables:
    float de(de) ;
        de:units = "meters" ;
        de:long_name = "DEPTH  (M)" ;
        de:generic_name = "depth" ;
        de:epic_code = 0s ;
    long minyear(time) ;
        minyear:units = "years" ;
    long maxyear(time) ;
        maxyear:units = "years" ;
    float bottom(time, lat, lon) ;
        bottom:units = "meters" ;
        bottom:long_name = "Bottom Depth" ;
    float pr(time, lat, lon, de) ;
        pr:units = "dbars" ;
        pr:_FillValue = -999999.f ;
        pr:long_name = "pressure" ;
    float te(time, lat, lon, de) ;
        te:units = "degrees C" ;
        te:_FillValue = -999999.f ;
        te:long_name = "in situ temperature" ;
    float th(time, lat, lon, de) ;
        th:units = "degrees C" ;
        th:_FillValue = -999999.f ;
        th:long_name = "potential temperature: pref= 0." ;
    float sa(time, lat, lon, de) ;
        sa:units = "psu" ;
        sa:_FillValue = -999999.f ;
        sa:long_name = "salinity" ;
    float ox(time, lat, lon, de) ;
        ox:units = "ml/liter" ;
        ox:_FillValue = -999999.f ;
        ox:long_name = "oxygen" ;
    short pr_cnt(time, lat, lon, de) ;
```
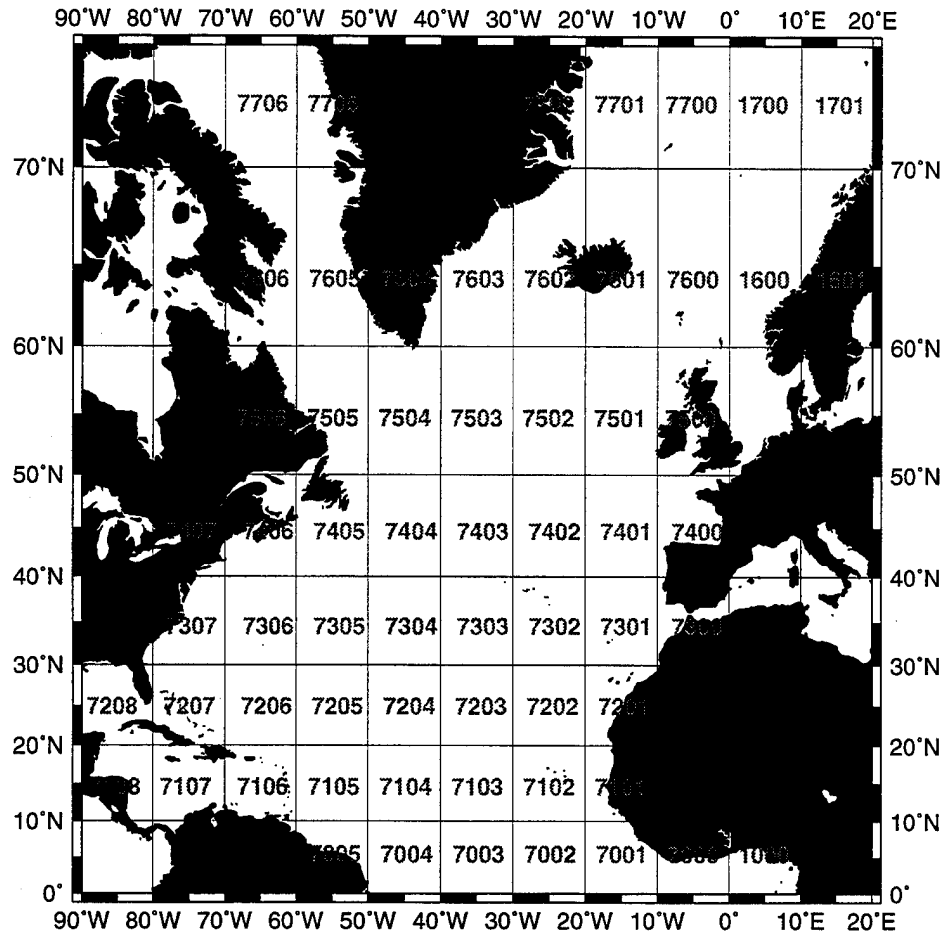
```
pr_cnt:_FillValue = 0s ;
     short te_cnt(time, lat, lon, de) ;
          te_cnt:_FillValue = 0s ;
     short th_cnt(time, lat, lon, de) ;
          th_cnt:_FillValue = 0s ;
     short sa_cnt(time, lat, lon, de) ;
          sa_cnt:_FillValue = 0s ;
     short ox_cnt(time, lat, lon, de) ;
          ox_cnt:_FillValue = 0s ;
     short de_cnt(time, lat, lon, de) ;
          de_cnt:_FillValue = 0s ;


// global attributes:
          :latmin = 10.f ;
          :latmax = 20.f ;
          :latincr = 0.5f ;
          :lonmin = -50.f ;
          :lonmax = -40.f ;
          :lonincr = 0.5f ;
          :node_offset = 1 ;
          :nprops = 6 ;
          :counts_included = 1 ;
          :title = "HydroBase" ;
          :command = "grid3d 7104 -D/scratch/ursa/ruth/natl2 -E.dat -B-50/-40/10/20 -I.5
-Ppr/te/th/sa/ox/s0 -Ctest.cdf -Ttime.bins" ;
          :compliance = "HydroBase  PMEL/EPIC  WHOI/OARS" ;
```

APPENDIX C:

10° WMO Squares: North Atlantic

90°W 80°W 70°W 60°W 50°W 40°W 30°W 20°W 10°W 0° 10°E 20°E

| | | 7706 | 77 | | | 7701 | 7700 | 1700 | 1701 |

70°N

| 06 | 7605 | | 7603 | 7602 | 7601 | 7600 | 1600 | |

60°N

| | 7505 | 7504 | 7503 | 7502 | 7501 | | |

50°N

| | 7406 | 7405 | 7404 | 7403 | 7402 | 7401 | 7400 |

40°N

| | 307 | 7306 | 7305 | 7304 | 7303 | 7302 | 7301 | |

30°N

| 7208 | 7207 | 7206 | 7205 | 7204 | 7203 | 7202 | 72 |

20°N

| 8 | 7107 | 7106 | 7105 | 7104 | 7103 | 7102 | |

10°N

| | | 005 | 7004 | 7003 | 7002 | 7001 | | 10 |

0°

90°W 80°W 70°W 60°W 50°W 40°W 30°W 20°W 10°W 0° 10°E 20°E

DOCUMENT LIBRARY

Distribution List for Technical Report Exchange – May 1995

University of California, San Diego
SIO Library 0175C
9500 Gilman Drive
La Jolla, CA 92093-0175

Hancock Library of Biology & Oceanography
Alan Hancock Laboratory
University of Southern California
University Park
Los Angeles, CA 90089-0371

Gifts & Exchanges
Library
Bedford Institute of Oceanography
P.O. Box 1006
Dartmouth, NS, B2Y 4A2, CANADA

Commander
International Ice Patrol
1082 Shennecossett Road
Groton, CT 06340-6095

NOAA/EDIS Miami Library Center
4301 Rickenbacker Causeway
Miami, FL 33149

Research Library
U.S. Army Corps of Engineers
Waterways Experiment Station
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

Institute of Geophysics
University of Hawaii
Library Room 252
2525 Correa Road
Honolulu, HI 96822

Marine Resources Information Center
Building E38-320
MIT
Cambridge, MA 02139

Library
Lamont-Doherty Geological Observatory
Columbia University
Palisades, NY z10964

Library
Serials Department
Oregon State University
Corvallis, OR 97331

Pell Marine Science Library
University of Rhode Island
Narragansett Bay Campus
Narragansett, RI 02882

Working Collection
Texas A&M University
Dept. of Oceanography
College Station, TX 77843

Fisheries-Oceanography Library
151 Oceanography Teaching Bldg.
University of Washington
Seattle, WA 98195

Library
R.S.M.A.S.
University of Miami
4600 Rickenbacker Causeway
Miami, FL 33149

Maury Oceanographic Library
Naval Oceanographic Office
Building 1003 South
1002 Balch Blvd.
Stennis Space Center, MS, 39522-5001

Library
Institute of Ocean Sciences
P.O. Box 6000
Sidney, B.C. V8L 4B2
CANADA

National Oceanographic Library
Southampton Oceanography Centre
European Way
Southampton SO14 3ZH
UK

The Librarian
CSIRO Marine Laboratories
G.P.O. Box 1538
Hobart, Tasmania
AUSTRALIA 7001

Library
Proudman Oceanographic Laboratory
Bidston Observatory
Birkenhead
Merseyside L43 7 RA
UNITED KINGDOM

IFREMER
Centre de Brest
Service Documentation - Publications
BP 70 29280 PLOUZANE
FRANCE

50272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO. WHOI-96-01	2.	3. Recipient's Accession No.

4. Title and Subtitle	5. Report Date
HydroBase – A Database of Hydrographic Stations and Tools for Climatological Analysis	March 1996
	6.

7. Author(s) Ruth Gorski Curry	8. Performing Organization Rept. No. WHOI-96-01

9. Performing Organization Name and Address	10. Project/Task/Work Unit No.
Woods Hole Oceanographic Institution Woods Hole, Massachusetts 02543	11. Contract(C) or Grant(G) No. (C) OCE91-03364 (G) NA36GP0137 NA46GP0303

12. Sponsoring Organization Name and Address	13. Type of Report & Period Covered
National Science Foundation and the National Oceanic and Atmospheric Administration	Technical Report
	14.

15. Supplementary Notes

This report should be cited as: Woods Hole Oceanog. Inst. Tech. Rept., WHOI-96-01.

16. Abstract (Limit: 200 words)

This report documents the organization, functionality, and algorithms of a software package which operates as a database manager and toolset for climatological analysis of hydrographic station data. It details the methods of quality control used in construction of the small, but growing, database and discusses some of the improvements *HydroBase* methods offer over existing gridded databases, including a short comparison to Levitus's *World Ocean Atlas 1994* package. A large portion of this technical reference is devoted to describing the software modules and providing examples for their use.

17. Document Analysis   a. Descriptors

climatology
isopycnal averaging
hydrographic database

b. Identifiers/Open-Ended Terms

c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages 50
Approved for public release; distribution unlimited.	20. Security Class (This Page)	22. Price

(See ANSI-Z39.18)   *See Instructions on Reverse*   OPTIONAL FORM 272 (4-77)
(Formerly NTIS-35)
Department of Commerce